

MSX COM DISK DRIVE

2.^a Edição Revisada

MSX DISK BASIC
MSX DOS(=MS DOS)
HB-MCP(=CP/M 2.2)
ARQUIVOS:
SEQUENCIAL E
ALEATÓRIO



4
McGraw-Hill

McGraw-Hill

NÉLSON CASARI

MSX
com
DISK DRIVE



Valorize sua formação profissional,
seu futuro, sua consciência

MSX

com

DISK DRIVE

MSX DISC BASIC
MSX DOS (= MS - DOS)
HB - MCP (= CP/M 2.2)
ARQUIVO SEQUÊNCIAL
ARQUIVO ALEATÓRIO

2.^a edição

NÉLSON CASARI

McGraw-Hill
São Paulo
Rua Tabapuã, 1.105, Itaim-Bibi
CEP 04533
(011) 881-8604 e (011) 881-8528

Rio de Janeiro • Lisboa • Porto • Bogotá • Buenos Aires • Guatemala • Madrid • México • New York • Panamá • San Juan • Santiago

Auckland • Hamburg • Kuala Lumpur • London • Milan • Montreal • New Delhi • Paris • Singapore • Sydney • Tokyo • Toronto

MSX com DISK DRIVE

Copyright © 1989, 1987 da Editora McGraw-Hill, Ltda.

Todos os direitos para a língua portuguesa reservados pela Editora McGraw-Hill, Ltda.

Nenhuma parte desta publicação poderá ser reproduzida, guardada pelo sistema "retrieval" ou transmitida de qualquer modo ou por qualquer outro meio, seja este eletrônico, mecânico, de fotocópia, de gravação, ou outros, sem prévia autorização, por escrito, da Editora.

EDITOR: MILTON MIRA DE ASSUMPÇÃO FILHO

Coordenadora de Revisão: Daisy Pereira Daniel

Supervisor de Produção: José Rodrigues

Ilustrações: Marco Casari

Dados de Catalogação na Publicação (CIP) Internacional (Câmara Brasileira do Livro, SP, Brasil)

C33m Casari, Nélson, 1931
MSX com disk drive: MSX Disc basic, MSX DOS (=MS-DOS), HB MCP (=CP/M 2.2),
arquivo seqüencial, arquivo aleatório / Nélson Casari. -- São Paulo : McGraw-Hill, 1987.

Bibliografia.

1. MSX (Computadores) 2. Sistemas operacionais (Computadores) 3. Unidades de disco (Ciência da computação) I. Título.

87-0951

CDD-001.64
-001.6425
-001.6442

Índices para catálogo sistemático:

1. "Disk drive" : Computadores : Processamento de dados 001.6442
2. MSX : Computadores : Processamento de dados 001.64
3. Processadores de disco : Computadores : Processamento de dados 001.6442
4. Sistemas operacionais em disco : Processamento de dados 001.6425
5. Unidades de disco : Computadores : Processamento de dados 001.6442



30 243 653



S U M Á R I O

INTRODUÇÃO, pg.13

- 1 MEMÓRIAS AUXILIARES DO COMPUTADOR, pg. 15
- 2 DISK DRIVES OU PROCESSADORES DE DISCO, pg. 18
- 3 DISQUETES OU MINIDISCOS, pg. 21
 - 3.1 DISQUETE DE 5,25 POLEGADAS, pg. 21
 - 3.2 DISQUETE DE 3,5 POLEGADAS, pg. 22
 - 3.3 DISQUETES DE 8 POLEGADAS E
OUTROS DISCOS PARA COMPUTADORES, pg. 22
 - 3.4 FORMATAÇÃO DE DISQUETES, pg. 22
 - 3.5 DIRETÓRIO DE DISQUETES, pg. 24
- 4 INTERFACES, pg. 27
- 5 SISTEMAS OPERACIONAIS PADRÃO MSX PARA DISCOS, pg. 28
 - 5.1 MSX DISC BASIC, pg. 30
 - 5.2 MSX DOS, pg. 30
 - 5.3 HB-MCP, pg. 31
- 6 INSTALAÇÃO E ATIVAÇÃO DOS SISTEMAS
OPERACIONAIS PARA DISCOS, pg. 32
 - 6.1 INSTALAÇÃO DO SISTEMA OPERACIONAL PARA DISCO, pg. 32
 - 6.1.1 OPERAÇÃO E MANUSEIO, PG. 33
 - 6.2 ATIVAÇÃO DO MSX DISC BASIC, pg. 35
 - 6.3 ATIVAÇÃO DO MSX DOS, pg. 37

6.4	ATIVAÇÃO PARCIAL DO DISC BASIC E DO MSX DOS, pg. 39
6.5	ATIVAÇÃO DO HB-MCP, pg. 39
6.6	LIGANDO O COMPUTADOR SEM ATIVAR O SISTEMA OPERACIONAL PARA DISCO, pg. 40
7	OPERANDO COM UM OU DOIS DISK DRIVES, pg. 41
7.1	OPERAÇÃO COM DOIS DISK DRIVES, pg. 42
7.2	OPERAÇÃO COM APENAS UM DISK DRIVE, pg. 43
8	GENERALIDADES SOBRE MSX DISC BASIC, MSX DOS E HB-MCP, pg. 47
8.1	SINTAXE DOS COMANDOS, pg. 47
8.2	DOIS CARACTERES ESPECIAIS, pg. 50
8.3	CANCELAMENTO DE COMANDOS, pg. 52
8.4	CONVENÇÕES NA DESCRIÇÃO DOS COMANDOS, pg. 53
8.5	DISTINÇÃO ENTRE PROGRAMAS E ARQUIVOS, pg. 53
9	COMANDOS, INSTRUÇÕES E FUNÇÕES DO MSX DISC BASIC, pg. 55
9.1	TABELA DOS COMANDOS, INSTRUÇÕES E FUNÇÕES DO MSX DISC BASIC, pg. 55
9.2	DESCRIÇÃO DOS COMANDOS, INSTRUÇÕES E FUNÇÕES DO MSX DISC BASIC, pg. 56
	BLOAD, pg. 57
	BSAVE, pg. 59
	CLOSE, pg. 61
	COPY, pg. 62
	CVD, CVS, CVI, pg. 64
	DSKF, pg. 66
	DSKI\$, pg. 68
	DSKO\$, pg. 70
	EOF, pg. 72
	FIELD, pg. 73
	FILES/LFILES, pg. 76
	FÓRMAT, pg. 78
	GET #, pg. 79
	INPUT #, pg. 81
	INPUT\$, pg. 83
	KILL, pg. 84
	LINE INPUT #, pg. 85

	LOAD, pg. 86
	LOC, pg. 87
	LOF, pg. 88
	LSET, pg. 89
	MAXFILES, pg. 91
	MERGE, 92
	MKD\$, MKS\$, MKI\$, pg. 93
	NAME, pg. 96
	OPEN, pg. 97
	PRINT # E PRINT # USING, pg. 99
	PUT, pg. 102
	RSET, pg. 103
	RUN, pg. 104
	SAVE, pg. 105
	SYSTEM, pg. 108
	VARPTR, pg. 109
	VERIFY, pg. 110
9.3	PROGRAMA AUTO-EXECUTANTE - AUTOEXEC.BAS, pg. 111
9.4	OCORRÊNCIAS DE ERROS EM OPERAÇÕES DO MSX DISC BASIC E MENSAGENS DO SISTEMA, pg. 112
10	COMANDOS E RECURSOS DO MSX DOS, pg. 113
10.1	TABELA DOS COMANDOS DO MSX DOS, pg. 113
10.2	DESCRIÇÃO DOS COMANDOS DO MSX DOS, pg. 113
	BASIC, pg. 114
	COPY, pg. 115
	DATE, pg. 119
	DEL OU ERASE, pg. 121
	DIR, pg. 123
	FORMAT, pg. 125
	MODE, pg. 127
	PAUSE, pg. 128
	REM, pg. 129
	REN OU RENAME, pg. 130
	TIME, pg. 131
	TYPE, pg. 133
	VERIFY, pg. 134
10.3	AUTOEXEC.BAT - COMANDO AUTO-EXECUTANTE, pg. 135

10.3.1	INCLUSÃO DO AUTOEXEC.BAT NO DISQUETE, pg. 135
10.4	COMANDO COLETIVO OU PROCESSAMENTO EM LOTE, pg. 137
10.4.1	PARÂMETROS SUBSTITUÍVEIS OU "MANEQUINS" NOS COMANDOS COLETIVOS, pg. 140
10.5	CHAVES ESPECIAIS DE EDIÇÃO, pg. 144
10.5.1	EXEMPLOS DE USO DAS CHAVES ESPECIAIS DE EDIÇÃO, pg. 145
10.6	FUNÇÕES DE CONTROLE POR CARACTERE NO MSX DOS, pg. 146
10.7	OCORRÊNCIAS DE ERROS EM OPERAÇÕES DO MSX DOS E MENSAGENS DO SISTEMA, pg. 147
11	COMANDOS E RECURSOS DO HB-MCP, pg. 149
11.1	TABELA DOS COMANDOS DO HB-MCP, pg. 150
11.2	DESCRIÇÃO DOS COMANDOS DO HB-MCP, pg. 150
	AUTOLD, pg. 151
	BACKUP, pg. 153
	BATCH, pg. 154
	COPDOS, pg. 155
	COPY, pg. 157
	DIR, pg. 159
	DRINP E DROUT, pg. 160
	DSKCNV, pg. 162
	DUMP, pg. 165
	ERA, pg. 166
	FORMAT, pg. 167
	FUNÇÕES, pg. 169
	PIP, pg. 171
	REN, pg. 179
	SAVE, pg. 180
	STAT, pg. 181
	SUBMIT, pg. 186
	SYSGEN, pg. 189
	TERM, pg. 191
	TYPE, pg. 192
	USER, pg. 193
	XSUB, pg. 196
11.3	FUNÇÕES DE CONTROLE POR CARACTERE NO HB-MCP, pg. 197

- 11.4 OCORRÊNCIAS DE ERROS EM OPERAÇÕES
 DO HB-MCP E MENSAGENS DO SISTEMA, pg. 197
- 11.5 PECULIARIDADES DO HB-MCP, pg. 199

APENDICE, pg. 201

- A.1 GRAVAÇÕES NOS FORMATOS BINÁRIO, ASCII E LM, pg. 202
- A.2 ARQUIVOS DE ACESSO SEQUENCIAL OU ALEATÓRIO, pg. 206
 PROGRAMA GERENCIADOR DE
 ARQUIVO SEQUENCIAL, pg. 209
 PROGRAMA GERENCIADOR DE
 ARQUIVO ALEATÓRIO, pg. 212
- A.3 TABELA DAS CHAVES ESPECIAIS
 DE EDIÇÃO DO MSX DOS, pg. 216
- A.4 TABELA DAS FUNÇÕES DE
 CONTROLE POR CARACTERE DO MSX DOS, pg. 218
- A.5 TABELA DAS FUNÇÕES DE
 CONTROLE POR CARACTERE DO HB-MCP, pg. 219
- A.6 TABELA DE MENSAGENS DO MSX DISC BASIC, pg. 221
- A.7 TABELA DE MENSAGENS DO MSX DOS, pg. 225
- A.8 TABELA DE MENSAGENS DO HB-MCP, pg. 230
- A.9 TABELA PARA USO DO
 TECLADO DO EXPERT COM O HB-MCP, pg. 237
- A.10 ENTRADA DE DIRETÓRIO - MAPA DE
 SEQUENCIA E SIGNIFICADO DOS "BYTES", pg. 238
- A.11 BLOCO DE CONTROLE DE ARQUIVO - MAPA DE
 SEQUENCIA E SIGNIFICADO DOS "BYTES", pg. 239

BIBLIOGRAFIA, pg. 240

AGRADECIMENTOS A

ELEBRA INFORMÁTICA S.A.

que nos forneceu literatura técnica e, a preços subsidiados, uma IMPRESSORA MÓNICA EI 6011, na qual foi impresso todo o texto finalizado deste livro, e uma UNIDADE DE DISCO FLEXÍVEL EI 9410B, que serviu especialmente para estudos e ensaios com os sistemas operacionais MSX DISC BASIC e MSX DOS.

Os dois produtos foram testados e usados exaustivamente por nós, apresentando excelentes resultados.

EPCOM - EQUIPAMENTOS ELETRÔNICOS LTDA.

que nos forneceu literatura técnica e, a preços favorecidos, um MICROCOMPUTADOR HOTBIT HB-8000 e uma unidade operacional para disco, constituída por um "disk drive" DS/DD HB-6000 e por um conjunto interface-fonte de alimentação HB-3600, os quais serviram especialmente para estudos e ensaios sobre seu sistema operacional exclusivo HB-MCP.

Lamentavelmente, dado o fato de tal equipamento só nos ter sido fornecido quando este manual já se encontrava em fase final de elaboração, em meados de fevereiro de 1987, não pudemos testá-lo exaustivamente, o mesmo sucedendo com seu sistema HB-DOS. Todavia, sendo este de total compatibilidade com o MSX DOS, cujo ótimo desempenho satisfaz plenamente, não há dúvida de que oferece absoluta confiabilidade.

Outrossim, cabe-nos elogiar o ótimo desempenho do microcomputador HOTBIT HB-8000 e do sistema operacional para disco HB-MCP.

GRADIENTE ELETRÔNICA S.A.

que nos forneceu literatura técnica e, a preço subsidiado, uma unidade do MICROCOMPUTADOR EXPERT. Favoreceu-nos também, em demonstração, com uma interface para disco FDD 5 1/4 AND 3 1/2.

Os dois produtos nos permitiram, juntamente com o "disk drive" da ELEBRA, levar a cabo os estudos e ensaios exaustivos sobre os sistemas MSX DISC BASIC e MSX DOS, com excelente desempenho.

I N T R O D U C A O

Os comandos, instruções, funções e recursos dos três principais sistemas operacionais para disco utilizáveis nos computadores de padrão MSX somam mais de cem itens ou títulos, além das mensagens emitidas pelos sistemas e outros detalhes.

Alguns têm uso restrito e são memorizados facilmente. Muitos deles têm uso mais diversificado, oferecendo algumas opções quanto a recursos. Outros têm uso bastante diversificado e com múltiplas opções.

Guardá-los todos em nossa memória não é tarefa fácil, mesmo porque constituem apenas parte de um extenso elenco de comandos, instruções, funções e recursos que, juntos, compõem o universo de palavras ou nomes reconhecidos pelo computador e pelos diversos sistemas operacionais que o integram e que são usados em sua operação.

Assim, ao elaborar este manual, nosso intento foi reunir de forma objetiva e prática todos os recursos disponíveis nos três principais sistemas operacionais para discos utilizáveis nos computadores MSX, de forma a oferecer a seus proprietários um manual para consultas rápidas e elucidativas.

Todavia, restringirmos nosso trabalho a esse objetivo poderia ser eventualmente útil apenas para aqueles que já contam com todos os conhecimentos sobre tais sistemas, bem como sobre o equipamento e o material envolvido em seu uso, e que passariam a dispor de um manual para, apenas, refrescar a mente quando a necessidade momentânea de alguma informação ou detalhe não fosse prontamente atendida pelos recursos da memória espontânea. Em verdade, tais usuários constituem minoria em nosso meio e, podemos dizer, adquiriram seus conhecimentos a duras pe-

nas, pesquisando e praticando, buscando informações esparsas aqui ou acolá, muitas vezes em literatura estrangeira, já que sua disponibilidade em nosso mercado é ou era bastante reduzida, quase inexistente.

Visando, pois, facilitar a aquisição de noções básicas indispensáveis ao conhecimento e à compreensão dos sistemas em foco, precipuamente para aqueles que se iniciam na matéria, procuramos reunir nos capítulos iniciais deste manual, em forma tão concisa e prática quanto possível, as informações julgadas úteis para tal finalidade, incluindo descrições das principais memórias auxiliares de computadores, disquetes, "disk drives" ou processadores de discos, interfaces, instalação e ativação dos sistemas, operações com um ou dois "disk drives" etc.

Os três sistemas operacionais e seus comandos e recursos são descritos com todas as particularidades, tendo sido os comandos de cada um apresentados em ordem alfabética, para facilitar consultas.

Em apêndice localizado no final do livro, visando facilitar consultas, encontram-se informações sobre as modalidades de gravações de programas e arquivos em disquete.

Encontram-se também reunidas no apêndice as tabelas de chaves de edição, de funções de controle por caractere e de mensagens de cada um dos sistemas operacionais aqui abordados.

Ao longo do livro, diversos programas funcionais são apresentados, com a finalidade de elucidar aplicações dos comandos descritos.

No mesmo apêndice, encontra-se um capítulo especial sobre as duas principais modalidades de arquivos em disquete: arquivos de acesso sequencial ou aleatório. E, com a finalidade de evidenciar as diferenças fundamentais existentes entre ambos, são apresentados dois programas para utilização em "mala direta", os quais oferecem todos os recursos básicos para aplicações práticas e imediatas.

O AUTOR

1 - MEMÓRIAS AUXILIARES DO COMPUTADOR

De forma geral, os computadores necessitam de memórias auxiliares para otimizar seu uso.

Constituem memórias auxiliares de computadores - também denominadas memórias externas - os meios magnéticos e de estado sólido, citados a seguir, capazes de "guardar" por tempo indeterminado programas que podem ser inseridos na memória do computador sem a necessidade de digitação:

- discos rígidos e fitas magnéticas em carretéis, para computadores de grande porte e minicomputadores;
- minidisks ou disquetes e fitas cassetes, para mini e microcomputadores;
- cartuchos de circuitos integrados, para determinados tipos de computadores.

A razão de utilização de memórias auxiliares ou externas é a de substituir e complementar a memória RAM do computador, geralmente de capacidade limitada e volátil, pois os programas e registros ou dados existentes nela são perdidos ao ser desligado o computador.

Memórias auxiliares armazenam programas e dados por tempo indeterminado e quando seu conteúdo se torna dispensável, ou fora de uso, pode ser apagado, liberando o dispositivo para outros programas ou arquivos, o que o torna bastante acessível com relação a custo.

Um programa - qualquer que seja sua classificação - é transferido para uma memória auxiliar pelo processo de gravação, depois de digitado e testado no próprio computador.

Para tanto, os códigos e dados que o compõem na memória do computador são convertidos em sinais digitais específicos, ou em pulsos digitais no caso de cartuchos, e gravados nos meios citados.

No processo de gravação são usados gravadores de fita, processadores de discos (DISK DRIVES) - e gravadores de circuitos integrados.

Um programa armazenado em fita ou disco é inserido na memória do computador pelo mesmo dispositivo usado para sua gravação, mas em processo inverso, denominado carregamento.

Um programa gravado em cartucho é inserido automaticamente na memória do computador quando o cartucho estiver colocado num dos encaixes denominados SLOTS, existentes no computador, e este for ligado. É oportuno lembrar que um cartucho só deve ser colocado num SLOT do computador quando este estiver desligado, sob pena de ocorrer avarias em ambos.

Computadores de grande porte utilizam como memórias auxiliares discos rígidos de vários tamanhos e/ou fitas magnéticas em carretéis de grande capacidade.

Para minicomputadores usualmente são adotados discos rígidos e/ou disquetes de 5,25 e 8 polegadas.

No caso de microcomputadores, geralmente são memórias auxiliares disquetes de 3,5 ou 5,25 polegadas e/ou fitas cassetes. Alguns computadores só podem usar fitas cassetes como memórias auxiliares.

Os microcomputadores do padrão MSX podem usar simultanea ou alternativamente TRÊS TIPOS DE MEMÓRIAS AUXILIARES:

- disquetes de 3,5 ou 5,25 polegadas;
- fitas cassetes;
- cartuchos.

Para que funcionem como memórias auxiliares de microcomputadores, os disquetes devem ser processados por

SISTEMAS OPERACIONAIS PARA DISCOS.

Neste livro abordaremos os sistemas operacionais para discos específicos para computadores do padrão MSX.

Antes, porém, repassaremos as noções básicas imprescindíveis sobre processadores de discos (DISK DRIVES), disquetes e interfaces próprias dos sistemas.

2 - DISK DRIVES OU PROCESSADORES DE DISCOS

Normalmente denominado DISK DRIVE, ou simplesmente DRIVE, e também acionador ou controlador de disco, o processador de disco constitui-se em aparelho bastante versátil e complexo, pois que executa não somente gravações e reproduções de programas de computador.

Os processadores de disquetes são geralmente referidos como DRIVES ou DISK DRIVES. Assim, como deveremos referir-nos bastante amiudamente a tal periférico neste livro, usaremos em sua designação, quase sempre, apenas a expressão DISK DRIVE, sem aspas, em vez de acionador, controlador ou processador de disco ou disquete.

Em conjunto com INTERFACE apropriada, o DISK DRIVE controla precisamente a localização e o espaço ocupado no disquete por arquivos ou programas, bem como seus títulos, tipos e extensões em "bytes", de tal forma que, por meio de simples comando dado através do computador, podemos conhecer com impressionante rapidez todo o conteúdo de qualquer disco, com todos os dados citados, mais as datas respectivas de gravação e, ainda, o espaço remanescente no mesmo para novas gravações!

Um DISK DRIVE faz mais ainda:

- localiza por acesso direto qualquer programa ou arquivo existente no disco ou qualquer registro de um arquivo de dados;
- carrega na memória do computador e põe em execução ou, se desejarmos, apaga também qualquer programa, arquivo ou registro de arquivo existente no disco;
- altera ou substitui nomes de programas ou de arquivos do disco;
- anula, altera ou substitui registros de arquivos de dados;

- copia programas do e no mesmo disco, ou de um disco para outro;
- formata o disco, isto é, prepara-o adequadamente para desempenho das funções enumeradas.

Atualmente os DISK DRIVES são produzidos para processar disquetes de 3,5, 5,25 e 8 polegadas de diâmetro, como medidas padrões.

Há modelos que processam apenas uma face do disquete. São denominados

DISK DRIVES DE FACE SIMPLES.

Há modelos que processam automática e simultaneamente as duas faces do disquete e são denominados

DISK DRIVES DE FACE DUPLA.

Estes possuem dois cabeçotes de gravação/leitura e circuitos eletrônicos mais complexos. Por esta razão, custam mais do que os outros.

Alguns modelos armazenam os programas ou dados ocupando espaço de tal forma que são classificados como

DISK DRIVES DE SIMPLES DENSIDADE.

Outros modelos ocupam muito menos espaço no disco para armazenar programas ou dados - aproximadamente a metade em relação aos já citados - e são classificados como

DISK DRIVES DE DUPLA DENSIDADE.

De forma quase que geral, são adotados atualmente para operar com microcomputadores

DISK DRIVES DE 5,25 POLEGADAS.

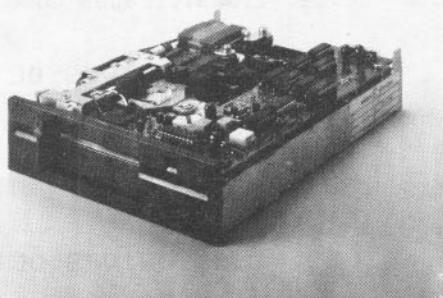
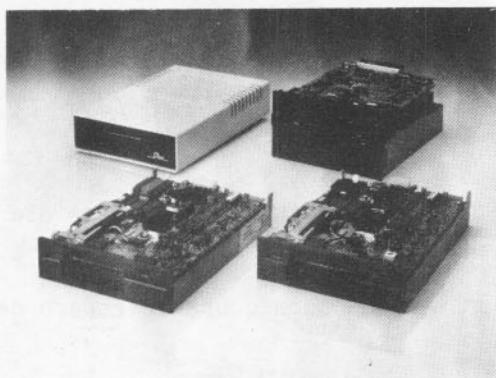
Os computadores da linha MACINTOSH requerem normalmente DISK DRIVES de 3,5 polegadas.

Os computadores do padrão MSX podem usar indiferentemente DISK-

DRIVES de 3,5 e de 5,25 polegadas.

DISK DRIVES de 8 polegadas geralmente são utilizados com minicomputadores.

Nas fotos reproduzidas a seguir são exibidos DISK DRIVES de 5,25 polegadas, sem estojo, e um com estojo.



3 - DISQUETES OU MINIDISCOS

Disquetes ou minidiscos - "floppy disks" - são discos de plástico flexível e de fina espessura, produzidos nas medidas de 3,5, 5,25 e 8 polegadas de diâmetro, além de outras medidas especiais.

Suas faces são revestidas de material magnético próprio para captar e reter, por processo de gravação, sinais produzidos e modulados sob padrões específicos para computadores.

O revestimento magnético do disquete varia em conformidade com o padrão de DISK DRIVE a que se destina seu uso.

3.1 - DISQUETES DE 5,25 POLEGADAS

FACE SIMPLES E DENSIDADE SIMPLES

Nessa modalidade, o disquete de 5,25 polegadas geralmente é provido de 35 trilhas magnéticas concêntricas, as quais comportam 10 setores cada. Cada setor tem a capacidade de armazenar 256 "bytes", sendo a capacidade total do disquete de 87,5 "kbytes" ($=35 \times 10 \times 256$).

FACE SIMPLES E DENSIDADE DUPLA

Nesse padrão, o disquete de 5,25 polegadas dispõe em cada lado de 40 trilhas, que podem ser divididas em 9 setores de 512 "bytes" ou 18 setores de 256 "bytes" cada, perfazendo uma capacidade total de arma-

zenamento de 180 "kbytes".

FACE DUPLA E DENSIDADE DUPLA

Sendo o padrão mais atual da medida de 5,25 polegadas, o disquete pode armazenar 360 "kbytes" de programas ou arquivos, dispondo em cada lado de 40 trilhas que podem ser divididas em 9 setores de 512 "bytes" ou 18 de 256 "bytes" cada, dependendo do sistema operacional ao qual é destinado.

3.2 - DISQUETES DE 3,5 POLEGADAS

Os disquetes de 3,5 polegadas são produzidos com características de FACE DUPLA E DENSIDADE DUPLA. São providos de 40 trilhas duplas em cada lado, as quais podem ser divididas em 9 setores de 512 "bytes" ou 18 setores de 256 "bytes" cada, perfazendo um total de 720 "kbytes" de capacidade de armazenamento de programas ou arquivos.

3.3 - DISQUETES DE 8 POLEGADAS E

OUTROS DISCOS PARA COMPUTADORES

São produzidas diversas modalidades para computadores e minicomputadores. Os discos rígidos e do tipo Winchester são contidos e operados em unidades seladas e sua capacidade de armazenamento é medida em "megabytes", havendo muitas variações nas capacidades disponíveis.

3.4 - FORMATAÇÃO DE DISQUETES

Antes de poder ser usado para armazenamento de programas ou arquivos, o disquete virgem ou novo de fábrica deve ser "formatado" de acordo com o sistema operacional em que vai ser utilizado.

A "formatação" é feita pelo próprio sistema operacional para dis-

co, no DISK DRIVE operante, através do comando FORMAT, e em conformidade com um programa "embutido" ou gravado na INTERFACE.

Se for usado um disquete não formatado para gravação de programas ou arquivos, o sistema emitirá uma mensagem de erro e o processo será abortado.

Tal problema é evitado quando, normalmente, os disquetes são formatados logo após sua aquisição, como primeira providência.

A formatação de disquetes é algo parecido como lotear uma área de terra, isto é, fazer um loteamento: a área é desembaraçada e dividida em quadras e estas em lotes. As quadras e lotes são demarcados e recebem códigos ou números de identificação para posterior referência e utilização (construção de moradias, armazéns, depósitos, escolas etc.).

No processo de "formatação", as trilhas são estabelecidas no disquete e divididas em setores. Trilhas e setores são "fichados" e codificados pelo sistema operacional para disco, de maneira que este possa localizá-los com exatidão para "arquivar" ordenadamente neles arquivos ou programas e poder encontrá-los posteriormente com rapidez e precisão, em conformidade com os comandos recebidos através do computador.

Para que todo esse processo se efetive com eficiência e precisão, existe no disquete um "diretório", como será visto logo mais.

Os padrões de formatação de disquetes variam de acordo com o sistema operacional adotado. Geralmente, computadores de uma mesma linha adotam um mesmo sistema operacional para disco, sendo então a formatação do disquete válida para diversas marcas.

Não obstante cada linha de computador (IBM, MSX, APPLE, TRS etc.) ter seu próprio sistema operacional para disco, é bastante comum ser adotado por muitos deles um mesmo padrão de disquete, próprio de um mesmo tipo de DISK DRIVE, já que a maioria dos produtores de computadores não produz seu próprio DISK DRIVE, existindo em realidade muitos fabricantes de computadores e relativamente poucos de DISK DRIVES.

Basicamente, a formatação de disquetes depende do DISK DRIVE e da INTERFACE DO SISTEMA OPERACIONAL. Esta não apenas compatibiliza o uso do computador com o DISK DRIVE, mas é responsável, também, pelo "tipo específico" de formatação a ser feita pelo DISK DRIVE no disquete.

No caso específico do MSX, que pode operar tanto com DISK DRIVE

de 5,25 como com o de 3,5 polegadas, a INTERFACE é produzida para formatar os dois tipos de disquetes, embora completamente diferentes.

Podemos dizer, com relação à formatação de disquetes, que o DISK DRIVE é a máquina que faz a operação e a INTERFACE a mentora da mesma.

Durante o processo de formatação, o disquete todo é testado pelo sistema quanto a possíveis defeitos. Se não os encontrar, a formatação é completada, não precisando jamais ser repetida, a não ser por determinadas conveniências.

Deve ter-se em conta, todavia, que, ao ser novamente formatado um disquete, todos os programas e arquivos contidos nele são apagados.

3.5 - DIRETÓRIO DE DISQUETES

Ao ser formatado o disquete, a uma determinada trilha é reservada uma função especial, a de receber o DIRETÓRIO DO DISQUETE, uma espécie de catálogo ou índice em que são registrados os nomes, tipos, extensão em "bytes", data de gravação e dados de localização de todos os arquivos e programas que são armazenados no disquete.

Nos sistemas DISC BASIC e MSX DOS, o DIRETÓRIO é situado na trilha inicial do disquete. Ao ser dado ao sistema um comando relativo a um arquivo ou a um programa, sua primeira ação é pesquisar no DIRETÓRIO sua existência e localização para, a seguir, acessá-lo e executar o comando.

O DIRETÓRIO é processado automaticamente pelo sistema operacional de disco, sendo atualizado sempre que um programa ou arquivo é gravado no disquete ou quando qualquer deles é apagado.

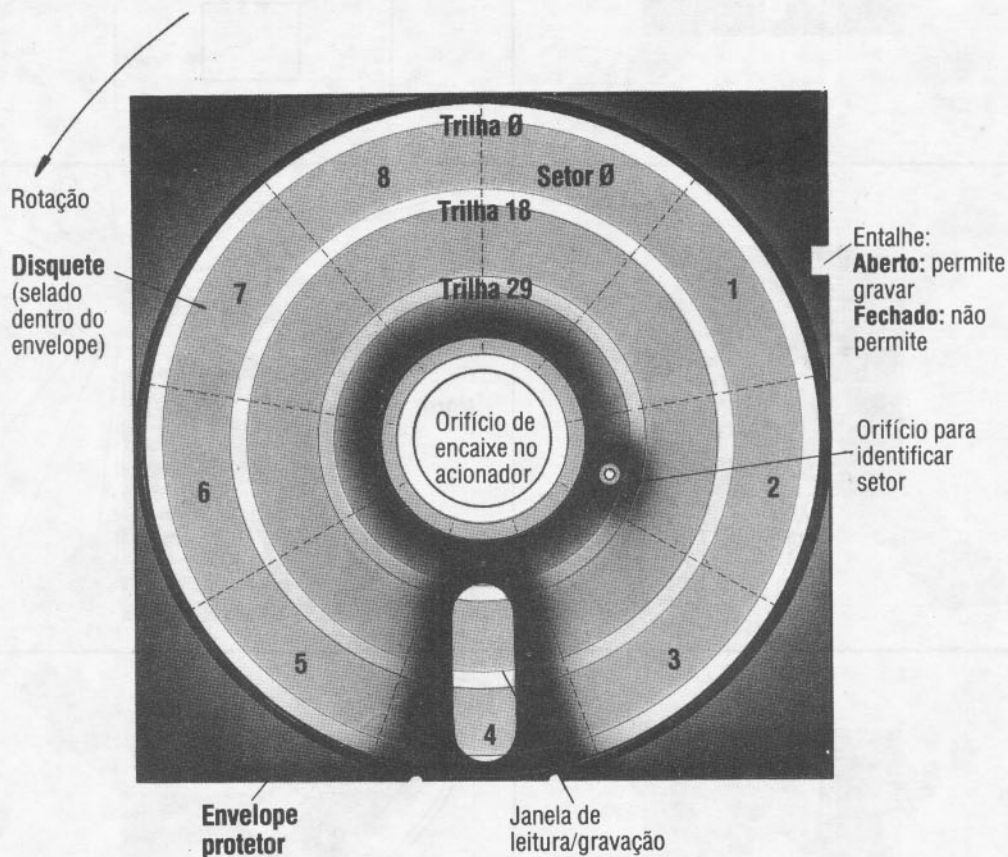
O DIRETÓRIO também é pesquisado pelo sistema operacional toda vez que, através de um dos comandos "files" ou "dir", é solicitada a lista dos programas ou arquivos contidos no disquete, solicitação essa normalmente feita através do computador pelo usuário, como logo veremos.

Não obstante o DIRETÓRIO dispor de espaço fixo bastante extenso para sua finalidade, pode acontecer que ele se esgote antes mesmo de a capacidade total de armazenamento do disquete ter sido utilizada. É o caso, por exemplo, de ser armazenada uma grande quantidade de arquivos

caso, por exemplo, de ser armazenada uma grande quantidade de arquivos e programas de reduzido tamanho - com poucos "bytes" - num disquete.

Em tal caso, poderá ser feito um remanejamento ou redistribuição dos programas e arquivos entre outros disquetes, para melhor aproveitamento do disquete em tal condição, ou desconsiderar o espaço ocioso.

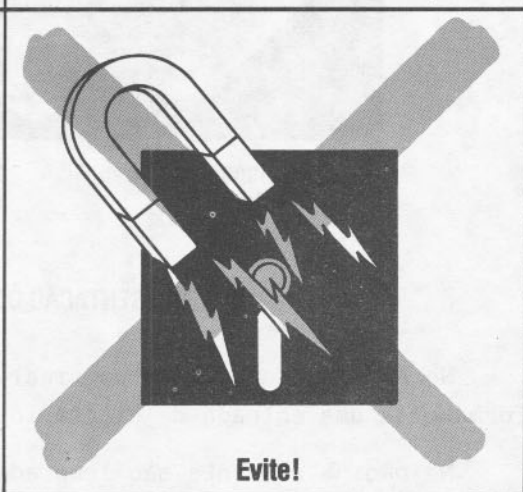
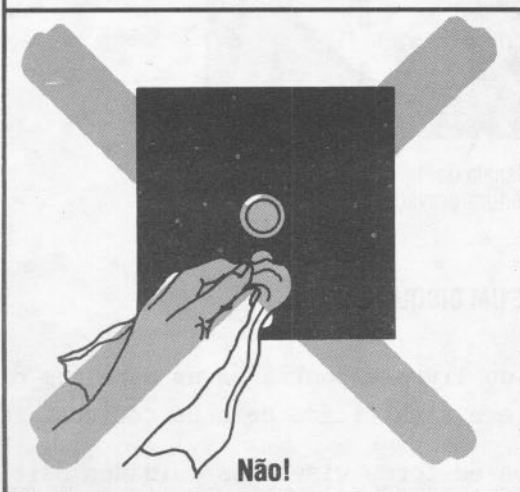
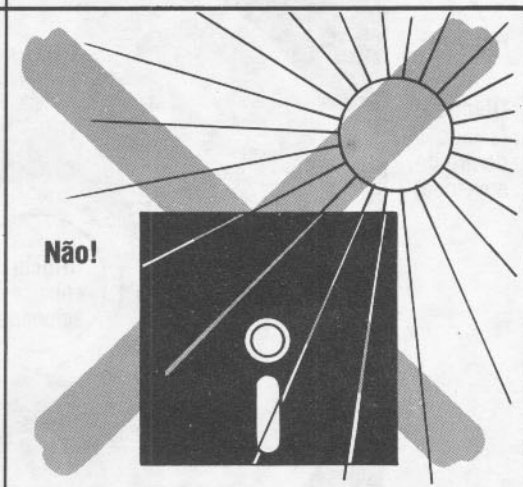
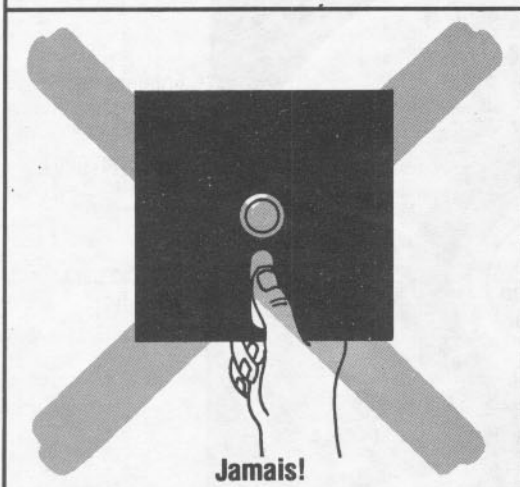
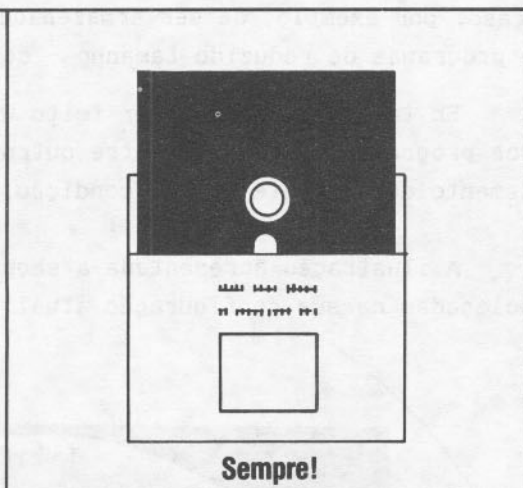
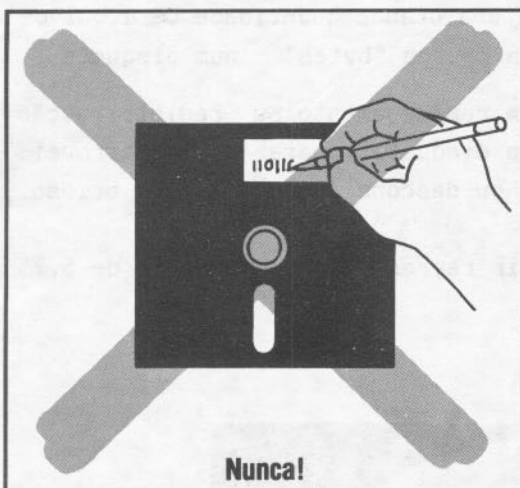
A ilustração apresentada a seguir representa um disquete de 5,25 polegadas na sua configuração atual.



REPRESENTAÇÃO DE UM DISQUETE

No Apêndice existente no final do livro encontra-se um mapa com o formato de uma entrada de diretório e o significado de seus códigos.

Na página seguinte são lembrados em forma visual os cuidados mais importantes a tomar com disquetes.



CUIDADOS A TOMAR COM DISQUETES

4 - INTERFACES

Na sua definição mais simples, INTERFACE é um dispositivo constituído por componentes e circuitos eletrônicos, destinado a interligar outros dispositivos ou elementos constituintes de um sistema.

Assim, a INTERFACE DO SISTEMA OPERACIONAL PARA DISCO é o dispositivo que interliga computador e DISK DRIVE, permitindo o funcionamento integrado e coordenado de ambos.

Além dessa função, a INTERFACE do sistema operacional tem outras.

Vimos no item 3.4 que um disquete precisa ser submetido a um processo de "formatação" para poder ser usado especificamente num sistema operacional para disco, sendo esse um processo de relevante importância, no qual a INTERFACE desempenha papel preponderante.

É também a INTERFACE que "sensoriza" se o sistema opera com uma ou duas unidades de DISK DRIVE, coordenando o processamento de disquetes em conformidade.

Identificar ou reconhecer o nome do programa que constitui o sistema operacional para disco especificamente válido para o conjunto em que está instalada e permitir seu carregamento na memória do computador é também função desempenhada pela INTERFACE.

Assim como o teclado e a tela de vídeo se constituem nos meios de comunicação entre o computador e seu usuário, uma INTERFACE de sistema operacional para disco é o elo de ligação e comunicação entre computador e DISK DRIVE, sendo os três - teclado do computador, tela de vídeo e interface do sistema operacional para disco - elos de ligação, ou, em amplo contexto, INTERFACES.

5 - SISTEMAS OPERACIONAIS

PADRÃO MSX PARA DISCOS

O padrão de computadores MSX adota três sistemas operacionais para processamento de disquetes:

DISC BASIC, MSX DOS e um compatível com CP/M (*)

(HB-MCP no HOTBIT) (*)

os quais permitem a utilização de DISK DRIVES e DISQUETES de 5,25 e 3,5 polegadas, sendo sua configuração de utilização de espaço:

- 5,25": 40 trilhas simples em um só ou em cada lado, sendo cada trilha dividida em 9 setores de 512 "bytes" cada um, constituindo uma capacidade de armazenamento de 180 ou 360 "kbytes";

- 3,5": 40 trilhas duplas em cada lado, sendo cada trilha dividida em 9 setores de 512 "bytes", constituindo uma capacidade de armazenamento de 720 "kbytes".

A formatação de disquete adotada para o DISC BASIC e MSX DOS é a do sistema MS DOS, enquanto que a utilizada pelo HB-MCP é a correspondente à do sistema CP/M, versão 2.2.

(*) MS DOS e MSX DOS são marcas registradas da Microsoft Corporation.

(*) CP/M e HB-MCP são marcas registradas da Digital Research e Epcom Equipamentos Eletrônicos Ltda., respectivamente.

Como vimos, funcionando como memória auxiliar do computador, o disquete deve receber e armazenar, através do processo de gravação,

programas da memória do computador e fornecê-los de volta, quando necessário, através do processo de reprodução. Deve também permitir efetuar cópias de programas nele mesmo e dele mesmo para outro disquete, bem como apagar programas ou mudar e alterar títulos e registros dos mesmos etc.

O processo de gravação e reprodução de programas no e do disquete e o processamento dos mesmos de forma geral são efetuados por

SISTEMAS OPERACIONAIS PARA DISCOS,

constituídos por:

INTERFACE DO SISTEMA e

PROCESSADOR DE DISCO OU DISK DRIVE

como componentes materiais ou "hardware", que operam em conjunto com o computador, e por

PROGRAMAS GERENCIADORES

os quais constituem as almas dos sistemas ou "software".

Para o MSX, estes programas gerenciadores - ou almas dos sistemas - são os seguintes:

- M S X D I S C B A S I C

- M S X D O S

- H B - M C P (HOTBIT)

Outros programas equivalentes ao HB-MCP existentes no mercado têm denominações diferentes.

5.1 - MSX DISC BASIC

O MSX DISC BASIC faz parte do sistema operacional próprio do computador. Podemos dizer que ele reside na memória permanente (ROM) do MSX.

Mas só se torna ativo, isto é, só se torna disponível ao usuário quando se instala no computador uma unidade processadora de disco juntamente com uma INTERFACE apropriada ao sistema.

Após ativado o MSX DISC BASIC, seus comandos e funções são operados igual e juntamente com os do MSX BASIC e controlam todas as operações de entrada e saída de programas em disco, além de gravar, reproduzir, apagar, copiar programas no mesmo disquete ou em outro, alterar ou mudar nomes de programas, indicar o conteúdo do disquete, o espaço disponível no mesmo etc.

Operando simultaneamente com o sistema operacional do computador, o DISC BASIC permite todas as suas operações normais, como, por exemplo, digitação, gravação ou carregamento de programas e sua execução, elaboração de cálculos, gráficos etc.

Permite também a operação normal com gravador cassete e quaisquer outros periféricos.

5.2 -- MSX DOS

O MSX DOS constitui sistema operacional não complementar ao BASIC MSX. É um sistema operacional EXCLUSIVO PARA DISCOS, desenvolvido com a finalidade de compatibilizar no MSX o processamento de programas e arquivos processados e tratados pelo MS DOS (MICROSOFT DISK OPERATING SYSTEM), o que significa que programas ou arquivos passíveis de processamento pelo MS DOS podem ser acessíveis ao MSX DOS.

O MSX DOS reside em memória externa - DISQUETE - precisando ser carregado na memória do computador para ser operado. Requer um mínimo de 64 "kbytes" de memória RAM para poder funcionar.

Entre outras funções, o MSX DOS "formata" disquetes para o sistema, copia programas num mesmo disco ou em outro, apaga-os, altera ou

muda nomes dos mesmos, indica quais os programas existentes no disquete, sua extensão em "bytes", data de gravação etc., mas não opera simultaneamente com o BASIC do sistema operacional do computador.

Para operar programas em BASIC, por exemplo, ou digitá-los, gravá-los e carregá-los da e para a memória do computador, seja de disco ou de fita cassete, é preciso invocar o BASIC, isto é, é necessário passar o controle do computador para o BASIC, o que deve ser feito através do comando específico do MSX DOS - comando denominado "BASIC".

O retorno do BASIC para o DOS processa-se também por meio de comando específico - neste caso do DISC BASIC - comando "CALL SYSTEM", como será visto oportunamente.

5.3 - HB-MCP

O HB-MCP também reside em disquete, precisando ser carregado na memória do computador para ser ativado e operado. Requer um mínimo de 64 "kbytes" de memória RAM para poder funcionar.

É constituído por diversos programas: os próprios do sistema e os de comandos internos e externos. (A diferença entre comandos internos e externos é que os primeiros estão "embutidos" no próprio sistema MCP e não podem ser visualizados pelo usuário, sendo apenas acessíveis ou disponíveis, enquanto que os comandos externos são como que "programas" do sistema e figuram no diretório do disquete que os contém, podendo ser visualizados através do comando DIR, como será visto oportunamente.)

O HB-MCP é uma versão do CP/M 2.2, de compatibilidade total com ele e desenvolvido especialmente para o computador HOTBIT. Quando ativado, opera sozinho, não permitindo acesso aos outros sistemas de discos nem ao BASIC.

Tanto o MSX DOS como o HB-MCP permitem que programas aplicativos como dBASE II (banco relacional de dados), SUPERCALC II (planilha eletrônica), WORDSTAR (processador de textos) e similares sejam utilizados nos computadores EXPERT e HOTBIT.

Veremos a seguir como instalar e ativar os três sistemas.

6 - INSTALAÇÃO E ATIVAÇÃO DOS SISTEMAS OPERA- CIONAIS PARA DISCOS

6.1 - INSTALAÇÃO DO SISTEMA

OPERACIONAL PARA DISCO

A instalação do sistema operacional para disco num computador MSX consiste em:

- encaixar em qualquer dos "slots" do computador o cartucho que contém a INTERFACE DO SISTEMA;
- encaixar o(s) conector(es) do cabo flexível que sai do cartucho da INTERFACE no(s) receptáculo(s) correspondente(s) do(s) DRIVE(s);
- ligar o(s) cabo(s) de alimentação de força (corrente contínua) da FONTE DE ALIMENTAÇÃO na(s) tomadas(s) própria(s) existente(s) no(s) DISK DRIVE(S);
- ligar na tomada de rede elétrica a FONTE DE ALIMENTAÇÃO, tendo cuidado para que a chave de voltagem (110 ou 220 volts) esteja na posição correspondente.

CUIDADOS A SEREM TOMADOS:

- todas as conexões devem ser feitas com os APARELHOS DESLIGADOS;
- não colocar o(s) DISK DRIVE(S) e disquetes junto ou próximos a campos magnéticos fortes. Monitor de vídeo e TV geram campos magnéticos intensos, sendo aconselhável manter DISK DRIVE e disquetes afastados pelo menos 20 cm dos mesmos.

6.1.1 - OPERAÇÃO E MANUSEIO

Sumariamente, indicamos a seguir os pontos mais importantes a observar na operação e manuseio do sistema operacional para discos:

- de forma geral, ao usar o computador com o sistema operacional para disco, a alimentação deste deve ser ligada antes e não concomitantemente com a daquele. Não é boa norma ligar o plugue da fonte de alimentação do DISK DRIVE na tomada existente na parte de trás do computador, já que esta é suprida de energia elétrica apenas quando é ligado o aparelho;

- é boa norma colocar o disquete no DISK DRIVE depois de este estar já ligado, mas antes de que o computador o seja;

- se for operado apenas o MSX DISC BASIC, não há necessidade de inserir antes no DISK DRIVE o disquete que vai ser processado, nem outro qualquer, antes de ser ligado o computador;

- se o conjunto vai ser operado com o MSX DOS ou HB-MCP, é necessário que, antes de ser ligado o computador, seja alojado no DISK DRIVE o disquete que contém os programas que constituem o sistema que vai operar. Caso o computador já esteja ligado, será necessário provocar um "reset" no mesmo, depois de alojado o disquete no DISK DRIVE.

COLOCAÇÃO DO DISQUETE NO DISK DRIVE:

- girar no sentido anti-horário a pequena alavanca existente no painel frontal do DISK DRIVE, posicionando-a horizontalmente;

- remover o disquete de seu envelope protetor, segurando-o levemente em plano horizontal e pelo lado da etiqueta, devendo esta estar sempre posicionada no lado de cima;

- com a janela ovalada de seu invólucro sempre voltada para a frente, introduzir o disquete no DISK DRIVE, fazendo-o deslizar livremente na fenda apropriada, até ser produzido o "clique" indicativo de seu encaixe correto. Cuidado deve ser tomado para não curvar ou dobrar o disquete, o que poderá inutilizá-lo;

- girar a alavanca do painel frontal do DISK DRIVE no sentido horário, posicionando-a verticalmente no seu limite.

REMOÇÃO DO DISQUETE DO DISK DRIVE:

- girar a alavanca do painel frontal do DISK DRIVE no sentido anti-horário. Um "clique" deverá ser produzido e o disquete será ejetado parcialmente do DISK DRIVE;

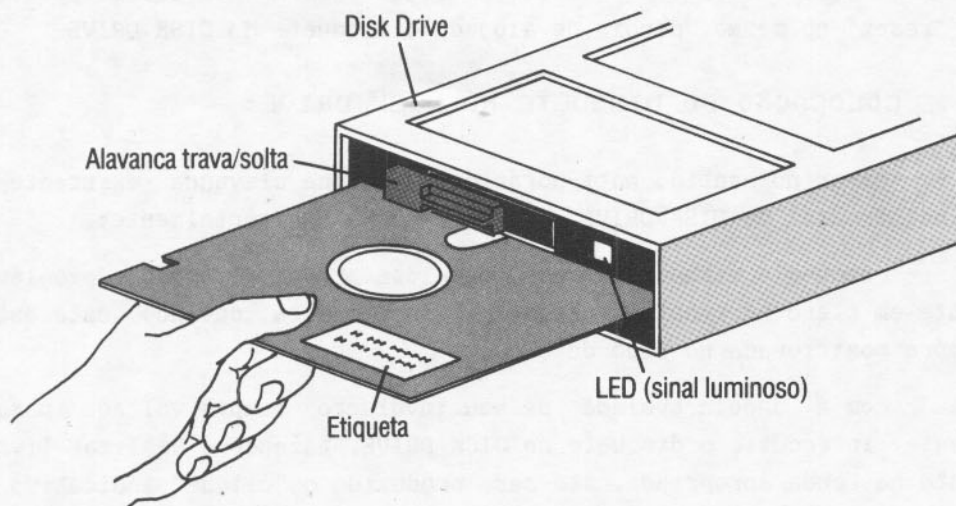
- apanhar o disquete com as pontas dos dedos e retirá-lo, tendo os mesmos cuidados adotados quando foi alojado no DISK DRIVE;

- colocá-lo logo em seguida em seu envelope protetor.

IMPORTANTE:

- nunca alojar ou desalojar disquetes no e do DISK DRIVE enquanto estiver acionado, isto é, em movimento. Se tal acontecer, as gravações existentes nos mesmos serão danificadas. Quando em movimento, a pequena luz vermelha de seu painel frontal permanece acesa.

Como observação final, recomenda-se que as instruções dos fabricantes dos aparelhos sejam atentamente seguidas para sua instalação, uso e manutenção.



COLOCAÇÃO/EXTRAÇÃO DO DISQUETE

6.2 - ATIVAÇÃO DO MSX DISC BASIC

Como vimos, o MSX DISC BASIC reside no próprio computador e seus comandos são liberados quando uma unidade de disco (composta por um ou dois DISK DRIVES + INTERFACE DO SISTEMA) é acoplada ao mesmo.

Assim, estando a INTERFACE devidamente encaixada num dos "slots" existentes no computador para essa finalidade e o(s) DISK DRIVE(S) conectado(s) a ela, o DISC BASIC é ativado ou inicializado automaticamente quando o computador é ligado.

Como o MSX DISC BASIC reside na ROM do computador, não é necessário que esteja alojado no DISK DRIVE o disquete que vai ser operado ou processado. Este poderá ser colocado no DISK DRIVE a qualquer momento.

É preciso, porém, que o DISK DRIVE esteja devidamente acoplado ao computador (através da INTERFACE DO SISTEMA) e que, ao ser este ligado, seja ou já esteja ligada a alimentação daquele.

Uma vez ativado o sistema, aparecerá no vídeo a mensagem:

Entre com data (D/M/A): (Hotbit)

Enter data (M-D-Y): (Expert/outros)

Para ser atendida, deverá a data ser digitada na ordem indicada, separando-se os dígitos com hífen (-), barra (/) ou ponto (.) e teclar RETURN em seguida. Caso não se deseje alterar a data corrente no sistema, basta pressionar uma vez a tecla RETURN.

Ato contínuo, isto é, pressionada a tecla RETURN ou digitada nova data, aparecerá no vídeo a etiqueta de abertura do sistema:

HOT-BASIC versão 1.1

EPCOM [Z.F.MANAUS] 1985

Mem. Livre 24455

DSK-BASIC V1.0

OK

Acima, a etiqueta exibida pelo HOTBIT. A seguir, a do EXPERT:

MSX BASIC versão 1.1. Br

Gradiente

23430 bytes livres

Disk BASIC version 1.0

OK

Note-se que, com a ativação do MSX DISC BASIC, a memória livre do computador passa a ser de 24.455 "bytes", em vez dos normais 28.815, se estiver sendo usada a INTERFACE produzida pela EPCOM para o HOTBIT HB-8000. Com a INTERFACE produzida pela GRADIENTE para o EXPERT, esse número é de 23.430 "bytes".

A razão dessa diminuição é que os "bytes" que constituem a diferença são destinados ao sistema operacional de disco, que necessita de uma área própria para funcionamento.

É preciso, pois, atenção no sentido de não tentar carregar na memória do computador programas que excedam, em extensão, aquele limite.

Se houver necessidade de processar (digitar e gravar ou carregar de fita cassete para o computador) programas maiores do que a quantidade disponível em "bytes" depois de acoplado e ativado o sistema operacional para disco - mas dentro do limite normal de 28.815 "bytes" -, será necessário ligar o computador sem ativar o sistema operacional para disco.

Para ligar o computador sem que seja ativado o sistema operacional para disco, basta manter a tecla SHIFT pressionada, ao ser acionada a chave de ligação, até que apareça no vídeo a etiqueta de abertura do MSX BASIC. O assunto é tratado no item 6.6.

Tais programas terão que ser mantidos em fita cassete, ou sofrer alguma redução, a menos que seja expandida a memória do computador.

Depois que aparecer no vídeo a etiqueta acima reproduzida, todos os comandos e funções do DISC BASIC estarão disponíveis para o operador, juntamente com os comandos, instruções e funções normais do BASIC MSX, bastando operá-los normalmente.

Deve ser observado que, ao ser ligado o computador, se não houver alojado no DISK DRIVE operante um disquete contendo o sistema DOS ou o HB-MCP, ainda que esteja ligada a alimentação do periférico, o processamento de discos poderá ser feito apenas através do DISC BASIC, sendo inacessíveis o MSX DOS ou o HB-MCP.

O acesso ao MSX DOS através do DISC BASIC só é possível na forma de retorno, isto é, quando o sistema tiver sido inicializado pelo MSX DOS e o controle passou para o DISC BASIC através do comando "BASIC" do próprio sistema. O retorno é feito pelo comando "CALL SYSTEM" e será válido apenas se o disquete que estiver no DISK DRIVE no momento de sua execução contiver os dois programas que compõem o MSX DOS.

Estando ativo o DISC BASIC, não é possível ter acesso ao HB-MCP, porque, como vimos antes, este é "exclusivista", não permitindo passar o controle para qualquer outro sistema, mesmo porque, como será visto oportunamente, a formatação do disquete no HB-MCP é diferente dos dois outros sistemas.

Deve ser observado também que, para utilização apenas pelo sistema DISC BASIC, é necessário somente que o disquete esteja "formatado", sendo comum o uso de discos que não contêm o MSX DOS.

6.3 - ATIVAÇÃO DO MSX DOS

Para ativar ou inicializar o MSX DOS, é preciso que o COMPUTADOR e o DISK DRIVE estejam conectados pela INTERFACE DO SISTEMA. É preciso também que o DISQUETE CONTENDO OS DOIS PROGRAMAS QUE COMPÕEM O SISTEMA esteja devidamente alojado do DISK DRIVE assumido. Como vimos antes, o MSX DOS reside em disquete e não na memória do computador.

A alimentação do DISK DRIVE deverá ser ligada antes de o computador entrar em funcionamento ou, pelo menos, simultaneamente.

Nessa condição, ao ser ligado o computador, os dois programas que compõem o MSX DOS serão automaticamente carregados no computador.

Ato contínuo, aparecerá no vídeo a etiqueta de abertura do sistema, reproduzida a seguir:

HB-DOS versão V1.6 (Hotbit)

[EPCOM]

HB-CCP versão V.1

Data corrente: Qua 1/01/1986

Entre c/nova data:

MSX DOS version 1.03 (Expert)

Copyright 1984 by Microsoft

Command version 1.11

Current date is Sun 1-01-1984

Enter new date:

A notação "Entre c/nova data:" ou "Enter new date:" induzirá o usuário a digitar a data atual. Caso não se deseje alterar a data corrente no disquete, bastará pressionar a tecla RETURN. Adotada uma das providências, aparecerá imediatamente no vídeo o sinal de prontidão do sistema - >A - seguido do cursor (sinalizado pelo habitual sinal de um quadrado), e todos os comandos do MSX DOS estarão disponíveis para o usuário. Frise-se: apenas os comandos do MSX DOS.

Para acessar os comandos do BASIC ou do DISC BASIC será preciso "sair" do DOS e "passar" para o BASIC, o que é feito através da utilização do comando "BASIC" do próprio MSX DOS.

Para retornar ao MSX DOS deverá ser dado o comando "CALL SYSTEM" do DISC BASIC.

Quando o MSX DOS é ativado, o mapeamento de memória RAM do computador é mudado por chaveamento automático do sistema, não permanecendo nela as rotinas do programa interpretador BASIC e outras, de sorte que a RAM fica praticamente toda disponível para o sistema operacional para disco. Por sinal, essa é a principal razão de o comando COPY do MSX

DOS efetuar cópias de programas de um disquete para outro com rapidez incomparavelmente maior.

Estando ativado o MSX DOS, não é possível utilizar o sistema HB-MCP. Como já mencionamos, este funciona independentemente dos dois outros sistemas.

6.4 - ATIVAÇÃO PARCIAL DO

MSX DISC BASIC E DO MSX DOS

Faz-se a ativação apenas parcial do DISC BASIC e do MSX DOS mantendo-se pressionada a tecla CTRL quando tem início o processo de ativação ou inicialização do sistema operacional para disco.

Com esse recurso, o sistema é condicionado a operar com apenas um DISK DRIVE físico e lógico, sendo eliminadas algumas de suas rotinas.

Obtém-se nessa modalidade uma liberação adicional, ou uma economia de memória, de aproximadamente 1.500 "bytes", que podem ser utilizados no processamento de programas cuja extensão demande mais espaço do que o permitido no modo normal de operação com disquetes.

Nessa condição, todavia, o comando COPY não é válido na sua forma plena, podendo ser utilizado apenas para copiar programas ou arquivos no próprio disquete, com nomes trocados ou alterados.

6.5 - ATIVAÇÃO DO HB-MCP

Para ser ativado ou inicializado o HB-MCP é preciso que o computador e o DISK DRIVE estejam conectados através da INTERFACE DO SISTEMA e que o DISCO CONTENDO OS PROGRAMAS QUE O COMPÕEM esteja devidamente alojado no DRIVE operante. Como vimos antes, o HB-MCP reside em memória externa: no disquete.

A alimentação do DISK DRIVE deverá ser ligada antes de o computador ser ativado, ou, ao menos, concomitantemente.

Nessa condição, ao ser ligado o computador, o disquete é acionado automaticamente e, pelo mesmo modo, os programas que compõem o sistema e os comandos internos do HB-MCP são carregados na memória do computador. Os comandos externos que integram o HB-MCP permanecem no disquete e qualquer deles se torna válido e atuante quando seu nome é digitado e a tecla RETURN é pressionada em seguida. Essa particularidade será vista no capítulo "COMANDOS E RECURSOS DO HB-MCP".

Completado o carregamento automático do HB-MCP, que é efetuado em alguns segundos, surge no vídeo a etiqueta de abertura do sistema:

```
HB-MCP [EPCOM] V1.0
```

```
>A
```

com o cursor piscando, e o sistema estará ativado, com todos os seus comandos, funções e recursos disponíveis para o usuário.

Da mesma forma que ocorre com o MSX DOS, quando o HB-MCP é ativado, o mapeamento da RAM do computador é alterado, não permanecendo nela as rotinas do programa interpretador BASIC e outras, ficando a memória do computador toda disponível para o sistema operacional de disco, o que torna o processamento dos disquetes extremamente rápido.

O HB-MCP funciona com total independência das rotinas do BASIC e da própria interface do sistema de disco, utilizando-as somente quando é carregado pela primeira vez. O acesso ao DISK DRIVE e outros periféricos é feito por chaveamento automático do próprio sistema.

6.6 - LIGANDO O COMPUTADOR SEM ATIVAR

O SISTEMA OPERACIONAL PARA DISCO

Quando se deseja operar o computador sem ativar o DISC BASIC, MSX DOS ou HB-MCP, basta manter-se a tecla SHIFT pressionada ao ser ligada a chave liga-desliga do computador, até que apareça a etiqueta normal de abertura para operação apenas através do BASIC. Esse recurso evita que se tenha que desconectar desnecessariamente a INTERFACE do sistema operacional para discos.

7 - OPERANDO COM UM OU DOIS DISK DRIVES

O DISC BASIC, o MSX DOS e o HB-MCP podem operar com um ou dois DISK DRIVES.

A interface que serve aos três sistemas é munida de sensores que detectam se qualquer deles está operando com um ou dois DISK DRIVES.

A utilização de dois processadores de discos torna qualquer sistema mais ágil, sendo aconselhável ou necessária para viabilizar operações de considerável volume com maior rapidez.

A utilização de um só DISK DRIVE exige apenas alguma atenção mais na execução de comandos como "COPY", "COPDOS", "BACKUP", sendo, contudo, perfeitamente viável e eficiente em todos os sentidos.

Como dissemos, ao ser ativado qualquer dos três sistemas operacionais, é detectado pela interface controladora se há um ou dois DISK DRIVES em operação no conjunto. Se houver um apenas, ele é assumido, identificado e operado automaticamente pelo sistema como DISK DRIVE A, ou "default A disk-drive". Se houver um segundo conectado, este será automaticamente identificado e operado pelo sistema como DISK DRIVE B.

Em qualquer dos casos, quando um comando dado não especifica o DISK DRIVE que deve ser acionado, qualquer dos sistemas assume que A é o indicado para a operação.

O DISK DRIVE assumido automaticamente, ou "default disk drive", pode ser mudado para B, através do MSX DOS e do HB-MCP.

Para efetuar a mudança, o procedimento a adotar é o seguinte: estando o cursor em prontidão para comando, digite B: (letra "B" seguida de dois-pontos) e pressione a tecla RETURN. A partir dessa providência, o DISK DRIVE a ser assumido automaticamente pelo sistema opera-

cional será o B, passando o DISK DRIVE A a operar somente quando especificado num comando dado.

Exemplificamos a seguir a providência citada:

A> (cursor em prontidão para comando)

A>B: (digitar B: e pressionar a tecla RETURN)

B> (cursor em prontidão para comando)

Quando é ativado através do MSX DOS, o DISC BASIC opera automaticamente o DISK DRIVE que foi assumido pelo outro sistema.

Vejamos a seguir exemplos de operações com dois DISK DRIVES e com apenas um DISK DRIVE.

7.1 - OPERAÇÃO COM DOIS DISK DRIVES

Uma das operações mais comuns e constantes que se tornam mais cômodas quando executadas com dois DISK DRIVES em qualquer dos três sistemas aqui abordados é a de copiar gravações de um disquete em outro.

Se o programa a ser copiado denominar-se HIPOT1.BAS, por exemplo, e estiver no disco alojado no DISK DRIVE assumido e este for o A, deverá ser dado um dos comandos indicados a seguir, em conformidade com o sistema operacional que estiver ativado no momento:

COPY "HIPOT1.BAS" TO "B" (DISC BASIC)

COPY HIPOT1.BAS B: (MSX DOS)

PIP B:=-A:HIPOT1.BAS (HB-MCP)

(Presume-se que no DISK DRIVE B esteja adequadamente alojado um disquete para receber a cópia.)

Se o DISK DRIVE assumido for o B, nos exemplos acima deverá figurar a letra A em lugar da B.

Depois de executado o comando, aparecerá no vídeo:

OK	(DISC BASIC)
1 Arquivo(s) copiado(s)	(HB-DOS/Hotbit)
1 files(s) copied	(MSX DOS/Expert)
A) ou *	(HB-MCP)

Observação: outros detalhes sobre os comandos "COPY" e "PIP" são examinados nos capítulos que tratam especificamente do assunto.

7.2 - OPERAÇÃO COM APENAS UM DISK DRIVE

COM O DISC BASIC E COM O MSX DOS:

Para copiar o programa HIPOT1.BAS de um disquete para outro com apenas um DISK DRIVE, assumido como A e estando alojado nele o disquete que contém o programa a ser copiado, o comando deve ser dado como se o sistema operasse com dois DISK DRIVES:

COPY "HIPOT1.BAS" TO "B"	(DISC BASIC)
COPY HIPOT1.BAS B:	(MSX DOS)

Logo após iniciar-se o processo de cópia, será expedida e impressa no vídeo a seguinte instrução para o operador:

Insira disco no drive B:	(Hotbit)
e pressione qualquer tecla	
Insert diskette for drive b:	(Expert)
and strike a key when ready	

o que significa que deverá ser desalojado do DISK DRIVE o disquete que contém o programa a ser copiado (nesse ponto já copiado na memória do computador, totalmente se sua dimensão for menor do que a do "buffer" utilizado para essa finalidade pelo próprio sistema, ou parcialmente

se o programa for mais extenso) e colocado em seu lugar o disquete que deverá receber a gravação da cópia. Chamemos este disquete de "B" e o que contém o programa a ser copiado de "A".

Feita a troca de disquetes e pressionada qualquer tecla do computador, o processo prosseguirá. Dependendo da extensão do programa ou arquivo em processamento, poderá findar nessa etapa ou estender-se por outras. Neste caso, surgirá nova instrução para troca de disquetes:

Insira disco no drive A: (Hotbit)

e pressione qualquer tecla

Insert diskette for drive A: (Expert)

and strike a key when ready

significando agora que deve ser desfeita a troca, isto é, deve ser extraído do DISK DRIVE o disquete "B" e colocado em seu lugar o disquete "A". Nesta situação, segundos depois nova troca será solicitada.

No caso de encerrar-se o processo, será emitida mensagem correspondente, como será visto adiante.

COM O HB-MCP:

No caso do HB-MCP, o comando deve ser dado sem especificar o DISK DRIVE (o comando COPY do HB-MCP opera exclusivamente com uma unidade):

COPY HIPOT1.BAS

O comando deve ser dado com o disquete que contém o sistema alojado no DISK DRIVE, pois o comando COPY do HB-MCP reside no disquete.

Logo após ser iniciado o processo de cópia, será exibida no vídeo a seguinte mensagem:

Coloque disquete origem e

Tecla <RETURN>

Tomada a providência, instantes após nova mensagem é expedida:

Coloque disquete destino e

Tecla <RETURN>

Dependendo da extensão da cópia a ser efetuada, poderá repetir-se a necessidade de trocar os disquetes, como acontece com os dois outros sistemas. Em tal caso, o sistema operacional exibirá no vídeo as mensagens correspondentes.

Em qualquer dos sistemas, ao findar o processo surgirá no vídeo a mensagem correspondente:

OK	(DISC BASIC)
1 Arquivo(s) copiado(s)	(HB-DOS/Hotbit)
1 files(s) copied	(MSX DOS/Expert)
Cópia completa	(HB-MCP/Hotbit)

Como vimos, a operação de cópia de programas com apenas um DISK DRIVE é executada pelo DISC BASIC e MSX DOS com simulação de dois DISK DRIVES, sendo que o único operante assume ora o lugar correspondente a "A", ora o correspondente a "B". Assim, ao findar o processo, o sistema operacional executou sua última ação como se estivesse operando o DISK DRIVE B. Como o DRIVE assumido é o A, qualquer comando dado a seguir - que não especifique B como DISK DRIVE a ser acionado - provocará a emissão da seguinte mensagem no vídeo:

Insira disco no drive A: (Hotbit)
e pressione qualquer tecla

Insert diskette for drive A: (Expert)
and strike a key when ready

em razão de que, para o sistema operacional, o único DISK DRIVE ope-

rante ainda é, nesse momento, o B e, ao receber novo comando, este deve ser normalmente dirigido ao DISK DRIVE assumido, devendo, portanto, haver outra troca de disquetes, significando para o sistema operacional que o DISK DRIVE tido como B passa a ser novamente A.

Nessa situação, poderá ser ou não efetuada a troca de disquetes. Porém, como o disquete remanescente no DISK DRIVE é aquele em que acabou de ser gravada a cópia, sendo, portanto, conveniente mantê-lo alojado para verificar se a mesma foi de fato executada (embora muito dificilmente ocorra o contrário), bastará pressionar uma tecla qualquer do computador e, em seguida, operar normalmente o sistema.

No caso do HB-MCP, o fato descrito por último não ocorre porque, ao findar o processo de cópia, o sistema operacional solicita automaticamente ao operador colocar no DISK DRIVE o disquete com sistema, ou seja, o disquete que contém os programas que compõem o sistema HB-MCP.

Um ponto importante a ser considerado:

A duplicação de programas ou arquivos através do comando "COPY" é muitíssimo mais rápida quando efetuada pelos sistemas MSX DOS e HB-MCP, comparavelmente com o DISC BASIC.

A razão desse fato é que o DISC BASIC compartilha a memória (RAM) do computador com o BASIC, restando uma área restrita destinada ao armazenamento transitório do conteúdo do arquivo ou programa em processamento de cópia, já que, antes de ser gravado em outro disquete como cópia, ele deve ser transferido para a memória do computador.

Assim, dependendo da extensão do programa ou arquivo a ser copiado, o processo poderá desdobrar-se em várias etapas de execução, conforme a quantidade de partições a serem feitas pelo sistema.

Com o MSX DOS e HB-MCP tal fato ocorre apenas quando a extensão da matéria a ser copiada excede 50 "kbytes", já que tais sistemas dispõem da RAM praticamente com exclusividade, em vista de esta ter outro mapeamento quando qualquer deles é ativado e de não permanecer nela o interpretador BASIC e outras rotinas.

Assim, quando há apenas um DISK DRIVE disponível no sistema, cópias de programas e arquivos devem ser efetuadas através do MSX DOS ou HB-MCP, podendo ser copiados vários deles numa só etapa.

8 - GENERALIDADES SOBRE MSX DISC BASIC, MSX DOS E HB-MCP

8.1 - SINTAXE DOS COMANDOS

Genericamente, a sintaxe dos comandos dos três sistemas operacionais tratados neste livro é a mesma, ressalvadas algumas pequenas diferenças, como veremos a seguir.

No DISC BASIC:

COMANDO "DISPOSITIVO:NOMEPRO" O/A

No MSX DOS:

COMANDO DISPOSITIVO:NOMEPRO O/A

No HB-MCP:

COMANDO DISPOSITIVO:NOMEPRO O/A

A principal diferença na formatação de comandos dos três sistemas é a obrigatoriedade de uso de aspas nos comandos do DISC BASIC.

Em alguns comandos do HB-MCP, a ordem de especificação de nomes é inversa à do DISC BASIC e do MSX DOS.

Indicamos a seguir as significações das palavras nos modelos acima indicados:

COMANDO:

Deve ser uma das palavras que representam os comandos do sistema operacional para disco em uso.

DISPOSITIVO:

Pode ser A: ou B: (a letra sempre seguida de dois-pontos), significando DISK DRIVE A ou DISK DRIVE B, quando operam dois no sistema, no caso do DISC BASIC.

Pode ser A: ou B:, CON, PRN, LST ou IMP, significando DISK DRIVE A ou DISK DRIVE B, CONSOLE (=VÍDEO E TECLADO) e IMPRESSORA, no caso do MSX DOS e HB-MCP, sendo que neste todos os nomes de dispositivos devem ser seguidos de dois-pontos, enquanto que no MSX DOS tal obrigatoriedade se restringe às letras que designam os DISK DRIVES.

NOMEPRO:

Pode ser qualquer palavra que se adote para identificar um programa ou arquivo, com ATÉ OITO CARACTERES, desde que não seja uma das incluídas na OBSERVAÇÃO 3 (a seguir) e que não contenha nenhum dos caracteres mencionados na OBSERVAÇÃO 2 (também a seguir).

O nome do programa é imprescindível em qualquer comando que seja individual, isto é, destinado a um só programa ou arquivo. Se foi gravado no disquete com indicação do tipo de programa, este deverá também ser especificado no comando.

Tipo de programa é uma extensão do nome, geralmente adotada com a finalidade de facilitar a identificação do programa.

Pode ser qualquer palavra com ATÉ TRES CARACTERES precedida de um ponto (.), desde que não seja uma das incluídas na OBSERVAÇÃO 3 e que não contenha nenhum dos caracteres mencionados na OBSERVAÇÃO 2.

A indicação de tipo de programa é opcional, mas, por ser de grande utilidade para o usuário na identificação de programas, é geralmente adotada.

O/A (OPÇÕES E ARGUMENTOS):

Podem ser:

- indicação opcional e específica de DISK DRIVE para determinada operação, quando há dois acoplados ao sistema;
- indicação de novo nome ou tipo de programa quando efetuada uma cópia ou alteração, ou para simples mudança no diretório do disquete;
- complementação de comandos com os argumentos chamados "chaves", tais como /A, /B, /P etc., ou opções propriamente ditas.

OBSERVAÇÃO 1:

CARACTERES QUE PODEM SER USADOS EM NOMEPRO:

A-Z	0-9	\$	\$	#	<
>	%	(	)	-	@
\	^	{	}	õ	!

OBSERVAÇÃO 2:

CARACTERES QUE NÃO PODEM SER USADOS EM NOMEPRO:

, . ; : + = / ? * " []

OBSERVAÇÃO 3:

PALAVRAS QUE NÃO PODEM SER USADAS

COMO NOMEPRO E NEM COMO SUA EXTENSÃO:

AUX - refere-se a dispositivos auxiliares de entrada ou saída, ou periféricos, podendo ser qualquer que não tenha seu nome reconhecido pelo sistema operacional. Os mencionados a seguir, por exemplo, têm seus nomes reconhecidos pelo sistema operacional.

CON - refere-se ao teclado do console, que funciona como dispositivo de entrada para certos comandos.

LST e PRN (também IMP no HB-MCP) - referem-se a impressora.

NUL - argumento nulo para qualquer comando ou dispositivo.

Tais palavras são usadas como opções de comandos.

Comandos e seus complementos podem ser digitados indiferentemente em letras maiúsculas ou minúsculas. Os nomes e tipos de programas são automaticamente gravados no disquete em letras maiúsculas.

Os comandos e seus complementos devem ser separados por delimitadores aceitos pelo sistema operacional em que são usados, como indicado a seguir:

- espaço (), aspas (") e vírgula (,), no DISC BASIC;
- espaço (), vírgula (,), ponto-e-vírgula (;), sinal de igualdade (=) e barra (/), no MSX DOS;
- espaço () para separar nomes; vírgula (,) para separar comandos; colchetes ([]) para separar opções, no HB-MCP.

Nos três sistemas, o ponto (.) é usado para separar nome e extensão e dois-pontos (:) para designar dispositivos físicos ou lógicos.

8.2 - DOIS CARACTERES ESPECIAIS

A fim de agilizar o trato de nomes de programas ou arquivos, dois caracteres desempenham funções especiais nos sistemas operacionais para discos: sinal de interrogação (?) e asterisco (*). Ambos são denominados caracteres-chave (em inglês "wild cards" = cartas-chave).

O sinal de interrogação usado no nome do programa ou arquivo e em sua extensão (tipo de programa ou de arquivo) indica que qualquer caractere pode ocupar aquela posição num comando dado. Por exemplo:

DIR ESTUDO?.BAS

listará no vídeo todos os programas ou arquivos existentes no disquete do DISK DRIVE assumido e que tenham sete letras no nome, sendo as seis primeiras ESTUDO e a última qualquer caractere: S, l, 5, 7 etc.

É um recurso de grande utilidade quando se pretende selecionar um grupo de programas existentes num determinado disquete e que compõem um livro, ou diversos arquivos sobre um mesmo assunto, gravados sempre sob determinado nome, com distinção apenas do último caractere, em que um número, por exemplo, diferencia os capítulos ou arquivos. Poderiam ser arquivos como CAPIT1, CAPIT2, CAPIT3 e assim por diante.

Podem ser usados diversos sinais de interrogação num mesmo nome de programa ou de arquivo. Por exemplo:

```
COPY "CH????.ARQ" TO "B:"
```

O asterisco (*) usado no nome do programa ou em sua extensão indica que qualquer caractere pode ocupar a posição em que se encontrar e as seguintes, no nome ou na extensão, no DISC BASIC. Por exemplo, um comando dado na forma exemplificada a seguir:

```
FILES "HI*.BIN"
```

listará no vídeo todos os programas ou arquivos existentes no disquete alojado no DISK DRIVE assumido e que tenham nomes que comecem por HI, sendo sua extensão .BIN.

No MSX DOS o asterisco é ainda mais poderoso. Por exemplo, se for usado no comando

```
DIR CAP*
```

listará no vídeo todos os programas e arquivos cujos nomes comecem por CAP, sendo os demais caracteres e as extensões quaisquer.

A designação *.* (asterisco, ponto e asterisco) feita num comando qualquer referir-se-á a todos os programas existentes num disquete. Por exemplo, se for usada num dos comandos

DEL *.* ou KILL "**.*"

apagará todos os programas e arquivos existentes no disquete alojado no DISK DRIVE corrente.

8.3 - CANCELAMENTO DE COMANDOS

A rapidez na execução de comandos nos sistemas operacionais para discos é tal que, muitas vezes, torna-se impossível cancelar ou interrompê-los, não obstante sejam acionados recursos, ou "anticomandos", disponíveis para tal finalidade.

NO DISC BASIC:

- os comandos são interrompidos e/ou cancelados do mesmo modo que no BASIC. Por exemplo, dado o comando:

FILES

a tecla STOP, pressionada uma vez, detém a exibição dos nomes de programas ou arquivos no vídeo. Pressionada uma segunda vez, a exibição tem prosseguimento.

- se forem pressionadas simultaneamente as teclas CTRL+STOP durante a execução de um comando, este é cancelado ou abortado no ponto em que se encontrar;

- em alguns casos, pressionando-se simultaneamente as teclas CTRL+C, é também possível cancelar comandos;

NO MSX DOS E NO HB-MCP:

- alguns comandos e a instrução "pressione qualquer tecla" (ou "strike a key when ready") podem ser cancelados pressionando-se de uma só vez as teclas CONTROL (ou CTRL) e C;

- quando um comando como TYPE, por exemplo, produzir o total

enchimento da tela de vídeo e o consequente "rolamento" para cima, este pode ser detido pressionando-se CONTROL e S simultaneamente, e continuado após pressionar-se qualquer tecla.

Nos itens "FUNÇÕES DE CONTROLE POR CARACTERE NO MSX DOS" e "FUNÇÕES DE CONTROLE POR CARACTERE NO HB-MCP", 10.6 e 11.3, respectivamente, são descritos outros recursos auxiliares no controle de comandos dos sistemas operacionais para discos.

8.4 - CONVENÇÕES NA DESCRIÇÃO DOS COMANDOS

Tendo examinado as noções básicas dos sistemas operacionais MSX para discos, passamos agora a descrever os comandos, instruções e funções de cada sistema, e seus principais recursos.

Todos os comandos, instruções e funções serão exemplificados com relação à sua sintaxe ou formatação, de sorte que, visando facilitar a sua compreensão, adotaremos a seguinte convenção em sua apresentação:

- em vez de nome do programa ou arquivo e sua extensão, será usada simplesmente a palavra NOMEPRO em letras minúsculas;
- em vez de DISK DRIVE, será usada simplesmente a letra D seguida de dois-pontos e na forma minúscula (d:);
- dois colchetes retos, [], significarão que o contido neles é opcional quanto à sua inclusão no comando.

8.5 - DISTINÇÃO ENTRE PROGRAMAS E ARQUIVOS

Na literatura especializada, programas convencionais para computadores e arquivos de dados armazenados em disquete são genericamente denominados "arquivos", causando alguma confusão em certos casos.

Em verdade, há uma nítida distinção entre ambos:

- um programa propriamente dito - em BASIC ou em linguagem de máquina - pode ser gravado no disquete DIRETAMENTE pelo comando SAVE ou BSAVE. Pode também ser recuperado DIRETAMENTE do disquete por meio do comando LOAD ou BLOAD;
- um arquivo de dados só pode ser gravado no disquete e recuperado do mesmo INDIRETAMENTE, através de um programa gerenciador.

Para gravação, o programa gerenciador utiliza os comandos:

PRINT # quando o arquivo é sequencial, e

PUT # quando o arquivo é de acesso direto.

Para recuperação dos dados, o programa gerenciador utiliza os comandos

INPUT # ou LINE INPUT # para arquivo sequencial, e

GET # quando o arquivo é de acesso aleatório.

(Leia no Apêndice "ARQUIVOS DE ACESSO SEQUENCIAL OU ALEATÓRIO".)

Alguns autores estrangeiros utilizam mais apropriadamente as denominações:

ARQUIVO DE PROGRAMA

e ARQUIVO DE DADOS

quando se referem às gravações de ambos em disquete.

De nossa parte, julgamos suficiente adotar simplesmente as denominações específicas de:

PROGRAMA

e ARQUIVO

para nos referirmos a um e outro respectivamente, já que, a nosso ver, as palavras em si são bastante claras quanto às suas acepções.

9 - COMANDOS, INSTRUÇÕES E FUNÇÕES DO MSX DISC BASIC

9.1 - TABELA DOS COMANDOS, INSTRUÇÕES E FUNÇÕES DO MSX DISC BASIC

BLOAD (*)	FORMAT	MKI\$
BSAVE (*)	GET#	MKS\$
CLOSE (*)	INPUT# (*)	NAME
COPY	INPUT\$ (*)	OPEN (*)
CVD	KILL	PRINT# (*)
CVI	LINE INPUT# (*)	PRINT# USING (*)
CVS	LOAD (*)	PUT
DSKF	LOC	RSET
DSKI\$	LOF	RUN (*)
DSK0\$	LSET	SAVE (*)
EOF (*)	MAXFILES (*)	SYSTEM
FIELD	MERGE (*)	VARPTR (*)
FILES/LFILES	MKD\$	VERIFY

Observação: os asteriscos entre parênteses indicam comandos, instruções e funções comuns ao BASIC e DISC BASIC da linha MSX.

9.2 - DESCRIÇÃO DOS COMANDOS,

INSTRUÇÕES E FUNÇÕES DO MSX DISC BASIC

Para facilitar consultas, os comandos, instruções e funções integrantes do DISC BASIC serão apresentados em ordem alfabética, sempre no início de cada página, a partir da seguinte, com os nomes impressos em caracteres destacados.

Após os comandos, instruções e funções do MSX DISC BASIC, serão apresentados outros recursos do sistema.

O comando BLOAD é utilizado para carregar do disquete para a memória do computador programas em linguagem de máquina.

Sua sintaxe é:

```
BLOAD "[d:]nomepro",[R][S][des-  
locamento]
```

OPÇÃO R - Faz o programa entrar automaticamente em execução após seu carregamento.

OPÇÃO S - Carrega diretamente na VRAM (memória de vídeo) programas ou dados gravados da mesma pelo comando BSAVE com idêntica opção. Veja mais abaixo programa-exemplo.

DESLOCAMENTO - Essa opção é utilizada para alojar o programa que vai ser carregado em área diferente daquela em que foi gravado. Se utilizada, deve ser na forma de um número inteiro, positivo ou negativo.

Se não for especificado nenhum número para deslocamento, o programa será carregado nos mesmos endereços indicados no comando BSAVE quando foi gravado.

O deslocamento de área para carregamento só poderá ser utilizado em programas com endereços relocáveis, o que, infelizmente, não acontece com a maioria dos programas comercializados.

Exemplos de comandos:

```
BLOAD "A:HIPOT1.BIN"  
BLOAD "HIPOT2",R,S  
BLOAD "HIPOT3.LM",-1200
```

Segue um programa-exemplo para carregar na VRAM um desenho gravado da mesma com a OPÇÃO S:

```
10 SCREEN 2
20 BLOAD "RETANG.BIN",S
30 GOTO 30
```

(Veja em BSAVE o programa gerador do desenho, com o comando para gravação com a OPÇÃO S.)

O comando BSAVE grava em disquete um programa em linguagem de máquina existente na memória do computador.

Sua sintaxe é:

```
BSAVE "[d:]nomepro",endereço inicial, endereço final,[endereço para execução][,S]
```

Se houver dois DISK DRIVES operando no sistema e o disquete a receber a gravação estiver no que não é o assumido, d: deve ser especificado.

Os endereços inicial e final da área de memória em que está alojado o programa devem ser especificados, obrigatoriamente, podendo ser em números decimais ou hexadecimais.

A indicação do endereço para início de execução do programa é opcional. Se não constar do comando, será automaticamente assumido o endereço inicial em seu lugar.

OPÇÃO S - Faz com que o conteúdo da memória de vídeo (VRAM) seja gravado no disquete. Os seus endereços inicial e final devem ser especificados no comando. Constate a validade da OPÇÃO S com a seguinte experiência:

Digite o seguinte programa:

```
10 COLOR 6,11,4: SCREEN 2
20 LINE (60,40)-(200,100),2,BF
30 OPEN "GRP:" AS #1
40 DRAW "BM80,60": COLOR 1
50 PRINT #1, "ESTE É UM"
60 PRESET (110,80):PRINT #1,"RETÂNGULO"
70 DRAW "BM80,120"
80 PRINT #1,"OS ÂNGULOS DO"
90 PRESET (56,140):PRINT #1,"RETÂNGULO SÃO RETOS"
```

Em seguida, dê o comando

```
BSAVE "RETANG.BIN",0,16383,S
```

Depois de gravado o programa, apague-o da memória do computador e carregue-o do disquete novamente para a memória do computador, usando o comando BLOAD (consulte-o para detalhes).

Observação: deve ter-se em conta que, estando instalado no computador o sistema operacional para disco - e não havendo cartucho de expansão de memória instalado - o endereço da última posição disponível na RAM é 56951 com a INTERFACE produzida pela GRADIENTE, e 57976 com a INTERFACE produzida pela EPCOM, em consequência da ocupação parcial da parte alta da mesma pelo sistema operacional para disco.

Assim, no caso de programa em linguagem de máquina que vai ser inserido no computador por digitação ou carregamento de fita cassete, para ser gravado a seguir em disquete, deve ser cuidado para que seus endereços não ocupem área da RAM situada acima do endereço-limite mencionado, sob pena de desorganizar todo o sistema, obrigando a desligar e ligar novamente ("reset") o computador.

Programas em linguagem de máquina gravados em fita cassete e que, por ocasião do carregamento no computador, não permitem relocação na RAM, através da opção de deslocamento, devem ser carregados com o sistema operacional para disco desligado do computador (ver a respeito o item 6.6) se, parcial ou totalmente, ocuparem área situada além do endereço citado.

Para conhecer tal endereço, dê o seguinte comando ao computador:

```
PRINT PEEK(64586)+PEEK(64587)*256
```

Depois de um arquivo ter sido aberto pela instrução OPEN, para entrada ou saída de dados, e não havendo mais registros a gravar ou a ler, o arquivo deve ser fechado, a fim de que o processo se complete com a gravação de dados que ainda se encontrem no "buffer" do sistema. Os dados são transferidos para gravação no disquete apenas quando referido "buffer" lota ou quando o arquivo é fechado. É importante notar que o diretório do disquete é atualizado ao ser fechado o arquivo.

O comando usado para esse fim é CLOSE, com a seguinte sintaxe:

```
CLOSE [# n0], [# n0]
```

n₀ é o número de arquivo aberto pela instrução OPEN.

Diversos arquivos podem ser fechados sob um só comando.

Se não for especificado nenhum número no comando, todos os arquivos que estejam abertos no momento de sua execução serão fechados.

Além de CLOSE, há instruções que, quando executadas, fecham automaticamente arquivos abertos. São elas CLEAR, END, MAXFILES, NEW, RUN. STOP não se enquadra nesse tipo de ação.

A associação entre um determinado arquivo e um número encerra-se com o comando CLOSE. Assim, se #1 foi usado para um arquivo, por exemplo FIPES, que foi fechado, ato contínuo #1 pode ser usado na abertura do arquivo DATAPES ou outro qualquer.

Se, após o processamento de entrada de dados, um arquivo não for fechado pelo comando CLOSE, sua utilização posterior se torna impraticável, pois, além de não ser ele atualizado no diretório do disquete, são perdidos dados já digitados que eventualmente aguardam gravação e que são transferidos para o disquete somente quando o "buffer" que os aloja transitoriamente atinge o limite de sua capacidade de armazenamento ou quando é executado o comando CLOSE.

Por oportuno, lembramos que o número máximo de arquivos que podem ser abertos pelo sistema operacional para disco é seis.

O comando COPY efetua cópias de programas ou arquivos de um disquete para outro. Pode também duplicar um programa ou arquivo no mesmo disquete, se for dado um nome diferente à cópia.

Sua sintaxe é:

```
COPY "[d:]nomepro" TO "[d:][nomepro]"
```

Para copiar um programa em outro disquete, sem mudar o nome, não precisa ser indicado no comando o segundo nomepro, isto é, o que será dado à cópia, bastando indicar o dispositivo d:, conforme exemplos que seguem:

```
COPY "A:ABC.BAS" TO "B:"
```

```
COPY "TEXT0.ARQ" TO "B:"
```

Se houver apenas um DISK DRIVE operando no sistema, os comandos são dados na mesma forma de quando há dois. Sobre o assunto, consulte o Capítulo 7.

Os caracteres especiais ? e * podem ser usados para copiar diversos programas ou arquivos sob um único comando. Exemplos:

```
COPY "A:*.BAS" to "B:"
```

copiar de um disquete para outro todos os programas em que nomepro tiver como extensão .BAS.

```
COPY "A:PROG?.*" TO "B:"
```

copiar de um disquete para outro todos os programas ou arquivos cujos nomes tenham como quatro primeiros caracteres a palavra PROG, sendo o quinto caractere e a extensão quaisquer.

Podem ser usados vários sinais de interrogação em nomepro, como, por exemplo:


```
COPY "DA????.ARQ" TO "B:"
```

```
COPY "???NQ??.*" TO "B:"
```

Usando o caractere especial * em lugar do nome e da extensão, como no exemplo que segue:

```
COPY "*.*" to "B:"
```

todos os programas de um disquete serão copiados em outro.

Ocasionalmente, pode ser necessário duplicar um programa no mesmo disquete, para se proceder a alterações experimentais no mesmo, preservando-se o original intacto. Isto pode ser feito se for omitida no comando a indicação do dispositivo (d:) e alterado o nome do programa. Exemplo:

```
COPY "DATAPES.BAS" TO "DATPES1.BAS"
```

Na descrição das funções MKD\$, MKS\$ e MKI\$, pode-se ver que, para serem inseridos em arquivos de acesso direto, números ou variáveis numéricas devem ser convertidos em variáveis alfanuméricas, através daquelas funções.

Para recuperar do arquivo números convertidos e assim gravados, é necessário "desconvertê-los", isto é, fazer com que as variáveis alfanuméricas que os alojaram no arquivo sejam convertidas para seus valores numéricos ou variáveis numéricas correspondentes.

Essa é, precisamente, a função desempenhada por CVD, CVS e CVI.

Sua sintaxe é:

CVD (variável de campo
definida para 8 "bytes")

CVS (variável de campo
definida para 4 "bytes")

CVI (variável de campo
definida para 2 "bytes")

As variáveis de campo deverão ser as mesmas definidas pela instrução FIELD quando os dados foram gravados no disquete.

RESUMO:

- CVD converte o conteúdo de uma variável de campo de 8 "bytes" em um número de dupla precisão ou em uma variável numérica correspondente.
- CVS converte o conteúdo de uma variável de campo de 4 "bytes" em um número de simples precisão ou em uma variável numérica

correspondente.

- CVI converte o conteúdo de uma variável de campo de 2 "bytes" em um número inteiro ou variável numérica correspondente.

São denominadas variáveis de campo as variáveis definidas por uma instrução FIELD.

Exemplos:

```
X=CVD(ND$)+Y: PRINT X
Y=CVS(NS$): PRINT Y
PRINT CVI(NI$)
```

Programa-exemplo:

```
10 OPEN "A:FIPES.ARQ" AS #1 LEN=14
20 FIELD #1,8 AS DP$,4 AS SP$,2 AS IN$
30 GET #1,1
40 PRINT CVD(DP$)
50 PRINT CVS(SP$)
60 PRINT CVI(IN$)
70 CLOSE #1: END
```

Observação: o programa acima recupera dados arquivados pelo programa apresentado na descrição de MKD\$, MKS\$ e MKI\$.

É o comando utilizado para se conhecer a capacidade disponível disquete para armazenamento de programas ou arquivos.

Sua sintaxe é:

DSKF(nº)

podendo nº variar de 0 a 2 e significando:

- 0 - DISK DRIVE assumido ou por último acessado.
- 1 - DISK DRIVE A
- 2 - DISK DRIVE B

Dando-se o comando:

PRINT DSKF(0)

sobre um disquete recém-formatado, de FACE DUPLA/DENSIDADE DUPLA, surgirá no vídeo o número

354

indicando que o disquete dispõe de 354 "kbytes" para armazenamento de programas ou arquivos.

Deve ser observado que esse número é calculado pelo sistema operacional de acordo com os seguintes critérios:

- o disquete é dividido em 360 setores denominados lógicos, sendo cada setor lógico constituído por dois setores físicos de 512 "bytes" cada. Portanto, cada setor lógico dispõe de espaço equivalente a 1024 "bytes" e é sempre considerado por inteiro. Desse total, 6 são destinados a dados de formatação e ao diretório e 354 ao armazenamento de programas e arquivos;

- se um programa ou arquivo ocupar apenas 300 "bytes", por exemplo, todo um setor lógico será considerado ocupado pelo sistema operacional e um segundo programa de 500 "bytes" será por ele armazenado em outro setor lógico;

- se um programa ocupar 1400 "bytes", por exemplo, 1024 serão armazenados num setor lógico, os restantes 376 em outro e o sistema operacional diminuirá dois setores lógicos da disponibilidade de espaço no disquete.

O critério descrito é adotado para que o controle de conteúdo do disquete possa ser executado com precisão através de seu diretório.

No caso de disquete de FACE SIMPLES/DENSIDADE SIMPLES, cada setor lógico é constituído por apenas um setor físico de 512 "bytes", de maneira que um comando

PRINT DSKF(0)

sobre uma unidade recém-formatada retornará no vídeo o número

351

correspondente a 351 setores lógicos de 512 "bytes", ou seja: 179.712 "bytes" disponíveis para armazenamento de programas e arquivos.

É uma função do DISC BASIC que permite ler diretamente qualquer setor do disquete.

Sua sintaxe é:

DSKI\$ (n0,n0')

n0 é o número do DISK DRIVE em que se encontra o disquete, podendo ser:

- 0 - para o DISK DRIVE assumido ("default disk drive").
- 1 - para o DISK DRIVE A.
- 2 - para o DISK DRIVE B.

n0' é o número do setor do disquete.

Quando um setor do disquete é lido através da função DSKI\$, o seu conteúdo é alojado numa área de memória cujo endereço inicial é indicado pelos "bytes" 62289 (&HF351) e 62290 (&HF352).

Assim, a leitura do setor especificado é feita através da captação de seus códigos naquela área e de sua decodificação por intermédio da função CHR\$.

Utiliza-se a função DSKI\$ atribuindo-a a uma variável alfanumérica qualquer, na forma X\$=DSKI\$(n0,n0'), ou através da instrução PRINT, na forma PRINT DSKI\$(n0,n0').

Conforme visto no Capítulo 3, o disquete formatado para o padrão MSX dispõe de 40 trilhas em cada face e cada trilha é constituída por 9 setores de 512 "bytes", havendo num disquete 720 setores.

Em sua execução, a função DSKI\$ não verifica se o número de setor do disquete é válido ou não. Assim, se for indicado um setor não existente, 950 por exemplo, num programa de leitura, quaisquer dados existentes na área de memória referida serão captados como pertencentes ao

"setor fantasma" indicado.

O conteúdo dessa área, apontada pelos endereços 62289 e 62290, é alterado ou apagado ao ser executado algum comando do DISC BASIC, como OPEN, CLOSE, PRINT #, FILES etc.

O programa apresentado a seguir lê setores do disquete e apresenta seu conteúdo no vídeo:

```
10 CLS: CLEAR 1500: KEYOFF
20 INPUT "Setor a ler: "; S
30 PRINT DSKI$(0, S)
40 AREA = PEEK(62289!) + PEEK(62290!) * 256
50 FOR I = 0 TO 511
60 H$ = HEX$(PEEK(AREA + I)): H$ = RIGHT$("00" + H$, 2)
70 PRINT H$; " ";
80 IF POS(0) = 39 THEN PRINT
90 IF CSRLIN = 21 THEN 150
100 NEXT I: PRINT
110 PRINT TAB(3) "Para ler outro setor, tecle F5"
120 END
150 PRINT, "Aperte qualquer tecla p/continuar"
160 R$ = INPUT$(1): CLS: GOTO 100
```

Através da função DSK0\$ é possível escrever ou gravar diretamente em qualquer setor do disquete.

Sua sintaxe é:

DSK0\$ n0,n0'

n0 é o número do DISK DRIVE em que está alojado o disquete a ser processado.

n0' é o número do setor do disquete a receber a gravação.

Os dados que se deseja gravar devem ser alojados numa área livre de memória do computador e os "bytes" dos endereços 62289 (&HF351) e 62290 (&HF352) devem indicar ao sistema operacional seu ponto inicial, o que é feito através da instrução POKE, como exemplificado no programa abaixo apresentado.

Deve ter-se em conta que a gravação feita por esse processo não é cadastrada no diretório do disquete, de forma que sua leitura ou recuperação pode ser feita apenas através da função DSKI\$.

Outro ponto a ser cuidado é o que se refere ao setor em que será feita a gravação na forma direta, que, a nosso ver, tem utilidade pouco prática. Se, por alguma razão, for conveniente lançar mão desse recurso, convirá sempre pesquisar previamente o setor a fim de constatar se o mesmo não armazena dados que não devem ser destruídos, já que uma nova gravação num mesmo setor se sobrepõe à anterior, apagando-a.

Ainda assim, deve ter-se em conta que, se o disquete que contiver a gravação feita por esse processo continuar sendo utilizado para outras gravações através do sistema normal, será impossível evitar que a gravação feita através de DSK0\$ não seja superposta, parcial ou totalmente, e destruída, já que a alocação de dados no disquete é feita exclusivamente pelo sistema operacional com base no diretório.

Em disquetes cujo conteúdo não deve ser submetido a qualquer risco de dano, a utilização de DSK0\$ deverá ser evitada.

O programa apresentado a seguir se presta à utilização de DSKO\$:

```
10 CLEAR 1000, 55000!
20 E=55000!: C=0
30 FOR A=0 TO 511
40 B=VPEEK(A)
50 POKE E+C,B
60 C=C+1: NEXT A
70 FOR I=0 TO 2
80 READ D: POKE E+C+I,D
90 NEXT I
100 DATA 13,10,26
110 INPUT "Grava em qual setor";S
120 EA=INT(E/256)
130 EB=E-EA*256
140 POKE 62289!,EB
150 POKE 62290!,EA
160 DSKO$ 0,S
170 PRINT,,"Leia o setor";S;"com o programa apresentado
na descrição da função DSKI$"
```

Observação: após digitar e conferir o programa, pressione simultaneamente as teclas CTRL e L e digite o que desejar gravar nas doze primeiras linhas da tela de vídeo.

Em seguida, desloque o cursor para a linha seguinte e pressione a tecla F5. Aparecerá no vídeo a pergunta sobre qual setor deve ser feita a gravação, podendo ser digitado um número entre 12 e 719. Os setores compreendidos entre 0 e 11 são destinados a dados de formatação e ao diretório do disquete e não devem ser rasurados, sob pena de o sistema operacional "emperrar" quando utilizado tal disquete, obrigando a provocar um "reset" ou a desligar o computador e sistema operacional.

Cuidado deve ser tomado para não efetuar gravações em setores já ocupados, pois seu conteúdo será destruído!

EOF é uma função para utilização em leitura de arquivos sequenciais. Sua finalidade é verificar quando terminam os registros de um arquivo. Quando tal acontece, EOF assume o valor -1. Em caso contrário, assume o valor 0. Sua sintaxe é:

EOF (nº)

nº deve ser o número de arquivo aberto pela instrução OPEN.

EOF é utilizada na recuperação ou leitura de dados de arquivo com as instruções INPUT # e LINE INPUT #, a fim de evitar erro no processo, que ocorre quando a leitura de registros prossegue depois de terem eles sido esgotados.

Tal erro interrompe a execução do programa, emitindo a mensagem:

Fim do arquivo (Hotbit)

Input past end (Expert)

Programa-exemplo:

```
10 CLS: CLEAR 5000: MAXFILES=1: DIM N$(50)
20 OPEN "PIPES" FOR INPUT AS #1
30 C = 1
40 IF EOF(1)=-1 THEN 100
50 INPUT #1,N$(C)
60 PRINT N$(C)
70 C=C+1: GOTO 40
100 CLOSE #1: END
```

Observação: EOF na linha 40 pode ser usada também na seguinte forma:

```
40 IF EOF(1) THEN 100
```

A finalidade do comando FIELD é estabelecer e delimitar os campos de registros de arquivos de acesso direto, através de variáveis alfanuméricas. Sua sintaxe é:

```
FIELD [#]n0,q AS v1$,q AS v2$
```

onde n0 é o número do arquivo aberto pela instrução OPEN, q a quantidade de caracteres para cada variável e v1\$ e v2\$ variáveis alfanuméricas quaisquer, podendo haver tantas quantas forem necessárias.

Por exemplo, se um registro de arquivo for composto por dois campos - nome e endereço -, cada um com até 30 caracteres cada, a instrução FIELD deverá definir duas variáveis alfanuméricas de 30 "bytes" cada, ou seja, duas variáveis às quais podem ser atribuídos até 30 caracteres cada, o que é feito como no exemplo que segue:

```
FIELD #1,30 AS N$, 30 AS E$
```

Uma vez fornecidos ao computador os dados para cada campo, são os mesmos atribuídos às variáveis definidas por FIELD e gravados no disquete, constituindo um registro.

Tal processo, todavia, não se efetiva de modo direto, como veremos a seguir.

Para receber os dados que lhes serão atribuídos - correspondentes aos campos de cada registro - e poder passá-los para o disquete como "partições" do registro, as variáveis definidas por FIELD são alocadas numa área especial de memória do computador, denominada "buffer" de transição. Por essa razão, inclusive, não podem receber atribuição por meio das instruções LET, INPUT e LINE INPUT, sob pena de desorganizar o sistema todo, pois, em tal caso, os ponteiros relativos acusarão sua localização na área comum das variáveis alfanuméricas em vez de no seu lugar certo, que é o "buffer" de transição, especificamente criado para processamento de arquivos de acesso direto.

Assim, os dados são atribuídos primeiramente a variáveis normais,

através das instruções mencionadas, e, em seguida, transferidos para as especiais definidas por FIELD, através dos comandos LSET e RSET. Um exemplo é apresentado a seguir:

```
10 OPEN "FIPES" AS #1 LEN=60
20 FIELD #1,30 AS N$,30 AS E$
30 LET X$="BENEDITO ALEGRE"
40 LSET N$=X$
50 LET X$="RUA X Nº Y"
60 LSET E$=X$
```

...

Depois de atribuídos às variáveis definidas em FIELD, os dados são finalmente transferidos do "buffer" de transição para o disquete, através do comando PUT.

De forma geral, esse é o processo através do qual os registros são efetivamente "arquivados" no disquete.

No processo de leitura ou recuperação dos registros do disquete, a instrução FIELD deve ser a mesma utilizada para sua gravação, sendo a leitura feita pela instrução GET.

Através de GET é lido apenas um registro por vez, sendo seu conteúdo transferido para as variáveis definidas por FIELD e localizadas no "buffer" de transição. A transferência é feita de acordo com a capacidade em "bytes" definida para cada uma.

Estando no "buffer", os dados se tornam disponíveis pela instrução PRINT, para exibição no vídeo, ou por LPRINT, para a impressora.

A instrução FIELD só pode ser usada depois de ter sido aberto o arquivo pela instrução OPEN.

A fim de que cada registro ocupe no disquete apenas o espaço que lhe corresponde, a instrução FIELD deverá ser complementada pela função LEN, que estabelece o espaço exato a ser ocupado por cada registro no disquete, devendo ser especificado por ela um número que represente a soma dos espaços ocupados pelos campos que constituem o registro.

Se não for especificada por LEN a quantidade de "bytes" para cada registro, cada um deles ocupará automaticamente 256 "bytes" no disquete, correspondente ao tamanho do "buffer" de transição, o que poderá

significar considerável desperdício se o registro ocupar efetivamente um espaço bem mais reduzido.

Depreende-se do exposto, por outro lado, que a quantidade máxima de espaços a ser definida pela instrução FIELD não pode ser superior a 256 "bytes", não importando a quantidade de variáveis declaradas.

FIELD pode também ser usada em comando múltiplo num programa, como exemplificado a seguir:

```
10 OPEN "A:MALADIR.ARQ" AS #1 LEN=55
20 FIELD #1,25 AS NM$,30 AS EN$
30 FIELD #1,12 AS TL$,5 AS CP$,20 AS CI$,18 AS ES$
...
```

Programa-exemplo para arquivamento de registros:

```
10 OPEN "A:FIPES.ARQ" AS #1 LEN=64
20 FIELD #1,25 AS NM$,25 AS EN$, 14 AS CI$
30 FOR I=1 TO 3
40 READ N$,E$,C$
50 LSET NM$=N$
60 LSET EN$=E$
70 LSET CI$=C$
80 PUT #1,I
90 NEXT I: CLOSE #1: END
100 DATA JÚLIO SANTOS, R VIT MORAIS 719, SPAULO SP
110 DATA MARIA SANTOS, R STº ANTONIO 314, SANTOS SP
120 DATA CECÍLIA SILVA, R CURUÇA 112, JUNDIAÍ SP
```

Programa-exemplo para leitura de registros:

```
210 OPEN "A:FIPES.ARQ" AS #1 LEN=64
220 FIELD #1,25 AS NM$,25 AS EN$,14 AS CI$
230 FOR I=1 TO 3
240 GET #1,I
250 PRINT
260 PRINT NM$: PRINT EN$: PRINT CI$
270 NEXT I: CLOSE #1: END
```

Este comando propicia a visualização do conteúdo do disquete no vídeo.

Sua sintaxe é:

FILES "[d:][nomepro]"

Basta digitá-lo sem as opções (lembramos que as indicações entre colchetes são opcionais) e pressionar uma vez a tecla RETURN para que os nomes e respectivas extensões de todos os programas e arquivos contidos no disquete colocado no DISK DRIVE assumido sejam mostrados no vídeo em forma de lista.

Havendo dois DISK DRIVES ligados no sistema operacional, para se conhecer o conteúdo do disquete alojado no DISK DRIVE não assumido, d: deverá ser especificado no comando. Supondo que seja A o DRIVE corrente, o comando deve ter o seguinte formato:

FILES "B:"

Os caracteres especiais * e ? podem ser usados para selecionar tipos determinados de programas ou arquivos. Por exemplo, o comando

FILES "*.ARQ"

exibirá no vídeo os nomes de todos os programas ou arquivos do disquete que tenham a extensão .ARQ.

O comando

FILES "PROGRAM?.TXT"

exibirá no vídeo todos os nomes de programas ou arquivos cujas primeiras sete letras sejam as mesmas do comando e a oitava qualquer caractere que tenha sido usado para gravá-los, mais a extensão .TXT.

Se se desejar apenas confirmação no vídeo da existência de um determinado programa ou arquivo no disquete, deverá ser especificado no comando o nome do mesmo. Exemplo:

FILES "A:FIPES.BAS"

Se não existir tal programa no disquete, será emitida pelo sistema operacional, através do vídeo, a mensagem adequada:

Arq. não existe (Hotbit)

File not found (Expert)

LFLES

O mesmo que FILES, com a diferença apenas que os dados são enviados para a impressora em vez do vídeo.

Obviamente, o comando só será válido se houver uma impressora conectada ao computador, e estiver ligada.

Um dos comandos cuja necessidade de uso se impõe de imediato, depois de instalado o sistema operacional para disco e quando se deseja gravar programas em disquetes, é o comando `FORMAT`, cuja finalidade é preparar ou "formatar" nos padrões específicos do sistema o disquete, já que este sai virgem da fábrica.

Como vimos antes, os padrões de formatação de disquetes variam de acordo com os padrões ou linhas de computadores.

A sintaxe do comando é:

`CALL FORMAT` ou `_FORMAT`

Ao ser dado o comando, surgem no vídeo mensagens solicitando definições do operador sobre:

- se o comando deve ser executado no `DISK DRIVE A` ou `B`;
- se o `DISK DRIVE` a ser operado é de 3,5 ou 5,25 polegadas e se é de `FACE SIMPLES` ou `DUPLA`;
- se as trilhas são `SIMPLES` ou `DUPLAS`.

Estas mensagens variam de acordo com o produtor da `INTERFACE` utilizada no sistema operacional e podem ser emitidas em português ou inglês, mas seu conteúdo básico é o mesmo.

Digitadas as informações, que devem ser condizentes com o equipamento instalado, será iniciado o processo de formatação do disquete e, ao ser concluído, surgirá no vídeo a mensagem correspondente.

Nessa condição, o disquete estará desimpedido e poderá receber e armazenar programas ou arquivos pelo processo de gravação, através dos comandos `SAVE` ou `BSAVE`, ou por transferência de um para outro disquete, através do comando `COPY`.

No comando `FORMAT` do `MSX DOS` são indicadas as características de formatação de disquetes para os sistemas `DISC BASIC` e `MSX DOS`.

GET é um comando exclusivo do processamento de arquivos de acesso direto. Sua finalidade é ler ou recuperar registros de arquivo em disquete e colocá-los no "buffer" de transição do sistema, onde se tornam disponíveis através dos comandos do DISC BASIC ou do MSX BASIC.

Sua sintaxe é:

GET # n₀, n₀'

n₀ é o número sob o qual o arquivo foi aberto.

n₀' é o número do registro a ser lido. Se não for indicado, será lido o registro seguinte ao último comando GET executado.

O número máximo que pode ser adotado para um registro, e aceito pelo sistema, é 4.294.967.295.

Ao ser lido no disquete pelo comando GET, o registro é transferido para as variáveis de campo, definidas pela instrução FIELD, no mesmo formato e na mesma disposição em que foram gravados. Em função desse fato, a instrução FIELD, que deve obrigatoriamente ser usada também na leitura de registros de arquivos de acesso direto, deve ser absolutamente idêntica à que foi utilizada na gravação dos mesmos registros.

Uma vez lidos pela instrução GET e passados para as variáveis de campo alocadas no mencionado "buffer", os registros se tornam disponíveis através da instrução PRINT, para exibição no vídeo, ou de LPRINT, para a impressora.

Além de FIELD, também a instrução OPEN deverá preceder o comando GET num programa de leitura, no mesmo formato e disposição adotados no programa respectivo de gravação de registros.

Programa-exemplo:

```
10 OPEN "A:DATAPES.ARQ" AS #1 LEN=30
20 FIELD #1,23 AS NM$,7 AS TL$
30 FOR I%=1 TO 100
```

```

40 GET #1,I%
50 PRINT NM$,TL$
60 PRINT
70 LPRINT NM$,TL$
80 LPRINT
90 NEXT
100 CLOSE #1: END

```

O programa que segue cria arquivo para teste do programa acima:

```

10 OPEN "A:DATAPES.ARQ" AS #1 LEN=30
20 FIELD #1,23 AS NM$,7 AS TL$
30 FOR I%=1 TO 5
40 READ X$: LSET NM%=X$: X$=""
50 READ X$: LSET TL%=X$: X$=""
60 PUT #1,I%
70 NEXT I%
90 DATA BATISTA QUEIRÓS,257-2712
100 DATA JOSÉ SARMENTO,444-3245
110 DATA ELIANA ALMEIDA,234-4567
120 DATA ANTONIO PEDREIRA,811-7890
130 DATA CLINEU CARDOSO,901-1232

```


A finalidade do comando INPUT # é ler dados de um arquivo sequencial em disquete e atribuí-los a variáveis definidas por ele.

Sua sintaxe é:

INPUT # n₀, variável ou variáveis

n₀ deve ser o número do arquivo, definido pela instrução OPEN.

Variável ou variáveis é a designação do nome ou dos nomes definidos para as variáveis que receberão os dados do arquivo através do comando INPUT #.

Em sua execução, INPUT # capta os dados do arquivo - pela leitura do disquete - assim como INPUT os recebe do teclado e os atribui a variáveis de sua definição.

Com relação à leitura de valores numéricos gravados no disquete, o primeiro caractere encontrado, que não seja do código de espaço, de retorno do carro ou de avanço de linha, é assumido como sendo o começo de um número. Os mencionados códigos, ou uma vírgula, significam para INPUT # o fim do número.

Em se tratando de variáveis alfanuméricas, também os códigos relativos a espaço, retorno de carro e avanço de linha são ignorados na leitura, sendo considerado como início de dados qualquer outro caractere. Se este caractere for o de aspas ("), os dados serão compreendidos como sendo os situados entre este sinal e um segundo idêntico. Por tal razão, dados correspondentes a variáveis alfanuméricas ("strings") não podem usar aspas para outra finalidade, como, por exemplo, destacar uma das palavras que compõem o conjunto. Se houver tal necessidade, os dados devem ser passados para o arquivo por meio da instrução LINE INPUT, no programa de gravação.

Se o primeiro caractere de uma variável alfanumérica não for o de aspas, então seus dados serão considerados finalizados com o surgimento de um dos códigos relativos a espaço, retorno de carro ou avanço de linha, ou após a leitura de 255 caracteres.

A execução do comando INPUT # é suspensa ao encontrar o código de fim de arquivo (EOF, &H1A).

Pelo exposto, deduz-se que a leitura de dados de arquivo sequencial através de INPUT # pode oferecer alguns problemas se, precedentemente, a gravação não foi efetuada com critérios adequados.

Por essa razão, e principalmente com relação à gravação de dados alfanuméricos, aconselha-se a utilização da instrução PRINT # na forma mais explícita e, sempre que possível, conjugada com a instrução LINE INPUT em vez de INPUT.

Outro ponto que deve merecer especial atenção na leitura ou recuperação de dados através do comando INPUT # é o que diz respeito à sequência de leitura, que deve ser absolutamente correspondente com a da gravação. Por exemplo, se os dados foram gravados na ordem A\$, B\$ e A, a leitura terá que obedecer à mesma ordem, sob pena de haver confusão na recuperação dos mesmos.

É oportuno lembrar também que, sempre que um programa para gravação de dados alfanuméricos em arquivo utilizou a instrução LINE INPUT, o programa correspondente de leitura deve utilizar LINE INPUT # em vez de INPUT #.

Programa-exemplo:

```
10 OPEN "FICON.ARQ" FOR INPUT AS #1
20 INPUT #1, A$: PRINT A$
30 INPUT #1, B$: PRINT B$
40 INPUT #1, A: PRINT A
50 IF NOT EOF(1) THEN 20
60 CLOSE #1
70 END
```

Em certas circunstâncias, pode ser necessário pesquisar dados de um arquivo cujo programa gerenciador não possuímos. Não sendo conhecidos os tipos de dados arquivados - numéricos ou alfanuméricos, a ordem em que foram gravados e sua extensão -, podemos recorrer a um comando do MSX DOS, se estiver disponível, ou utilizar a instrução INPUT\$, que lê em um arquivo de dados um caractere ou uma sequência deles por vez.

Sua sintaxe é:

```
INPUT$(nº de caracteres,# nº de arquivo)
```

Número de caracteres indica a quantidade de caracteres para leitura a cada execução do comando.

Número de arquivo é o mesmo de sua abertura pela instrução OPEN.

Depois de lidos os dados e exibidos no vídeo, sua análise poderá fornecer as informações e indicações necessárias.

INPUT\$ é válido para arquivos sequenciais e de acesso direto.

Programa-exemplo:

```
10 OPEN "MALADIR.ARQ" FOR INPUT AS #1
20 IF EOF(1) THEN 60
30 A$=INPUT$(1,#1)
40 PRINT A$
50 GOTO 20
60 CLOSE #1: END
```

Observação: as linhas 30 e 40 podem ser substituídas por uma só linha:

```
30 PRINT INPUT$(1,#1)
```

A finalidade do comando KILL é eliminar do disquete programas ou arquivos. Sua sintaxe é:

KILL "[d:]nomepro"

Para ser apagado, um arquivo de dados deve estar fechado quando o comando é dado. Se for dado o comando KILL para um arquivo aberto, será abortado o processo, surgindo no vídeo a mensagem:

Arquivo aberto (Hotbit)

File already open (Expert)

Os caracteres especiais * e ? podem ser usados para apagar diversos ou todos os arquivos e programas de um disquete com um só comando.

O dispositivo d: deverá ser especificado se o disquete que contém o(s) programa(s) ou arquivo(s) a ser(em) eliminado(s) encontrar-se no DISK DRIVE não assumido.

Exemplos de comandos:

KILL "A:CONTAL.ARQ"

KILL "CONTA?.*"

KILL "*.*)"

Observação: uma vez dado, o comando do terceiro exemplo apaga todos os programas e arquivos do disquete alojado no DISK DRIVE assumido sem aguardar confirmação do usuário, como é o caso do comando equivalente do MSX DOS. Assim, antes de usá-lo, convém o operador certificar-se se, de fato, todos os programas e arquivos do disquete em processamento pelo comando podem ser destruídos.

A instrução LINE INPUT # é utilizada para ler ou recuperar dados de arquivo sequencial em disquete.

Sua sintaxe é:

LINE INPUT # nº do arquivo,
variável alfanumérica

Com a instrução LINE INPUT # é possível ler uma linha inteira de dados - até 254 caracteres - sem as restrições de INPUT #.

Os dados lidos devem ser atribuídos a variáveis alfanuméricas.

Os únicos limitadores válidos para LINE INPUT # são os códigos de retorno de carro e avanço de linha, colocados automaticamente quando os dados são digitados e gravados. Outros limitadores como aspas, vírgula, ponto-e-vírgula, espaço etc. são ignorados. Seu único limite é a quantidade de caracteres, que não pode exceder 255.

LINE INPUT # é bastante útil quando cada linha de dados de um arquivo é dividida em campos, ou quando um programa em BASIC gravado no formato ASCII deve ser lido como dados por outro programa.

Programa-exemplo:

```
10 CLS: CLEAR 500
20 OPEN "DATAPES" FOR INPUT AS #1
30 FOR C=1 TO 100
40 IF EOF(1)=-1 THEN 80
50 LINE INPUT #1,D$
60 PRINT: PRINT D$
70 NEXT C
80 CLOSE #1: END
```

Permite carregar do disquete para a memória do computador um programa em BASIC previamente gravado no formato binário ou ASCII.

Sua sintaxe é:

```
LOAD "[d:]nomepro"[ ,R]
```

Nomepro deve ser exatamente o mesmo nome usado na gravação do arquivo ou programa, com extensão, se houver. É indispensável em se tratando de disquete, ao contrário de fita cassete.

Para carregar na memória do computador programas de fita cassete gravados nos formatos binário e ASCII, os comandos a serem usados são CLOAD e LOAD, respectivamente.

No caso de disquetes, tal distinção não é necessária, pois o sistema operacional para disco detecta e controla automaticamente a diferença entre os dois formatos.

No item "GRAVAÇÕES NOS FORMATOS BINARIO, ASCII E LM" encontram-se outros detalhes de interesse sobre o assunto. Por oportuno, entretanto, lembramos que a diferença existe apenas na gravação. Depois de carregados na memória do computador, os programas são alojados normalmente, isto é, sem distinção.

OPÇÃO R - Quando especificada no comando, o programa entra automaticamente em execução após o carregamento.

Exemplos de comandos:

```
LOAD "CATREC.BAS"
```

```
LOAD "A:ARQPES.TXT",R
```


LOC é uma função do DISC BASIC válida no processamento de arquivos de acesso direto e sequenciais.

Em arquivo de acesso direto indica o registro gravado ou lido por último através dos comandos PUT e GET.

Em arquivos sequenciais indica o número de partições de setores (blocos de 256 "bytes") gravados ou lidos desde que o arquivo a que se refere foi aberto. A indicação é feita em múltiplos de 256.

Sua sintaxe é:

LOC (nº do arquivo)

Se um arquivo de acesso direto foi aberto mas não houve nenhuma operação de entrada ou saída de dados, LOC fornecerá o valor 0.

Em arquivo sequencial, depois de aberto no modo "FOR OUTPUT", se nenhum registro foi gravado e for utilizada a função LOC, esta fornecerá o valor 0. Depois de gravados registros, fornecerá a quantidade de partições de setores alocados, em múltiplos de 256. Na modalidade "FOR INPUT", depois de aberto o arquivo, LOC fornecerá o número 256 se nenhum registro foi lido. Após lido algum, indicará as partições verificadas. Lido todo o arquivo, indicará os "bytes" ocupados por ele.

Programa-exemplo:

```
10 OPEN "A:LISTEL.ARQ" AS #1 LEN=32
20 FIELD #1,20 AS NMS,12 AS TL$: C=1
30 GET #1,C
40 PRINT LOC(1);NMS;TL$
50 IF LOC(1)=10 THEN 70
60 C=C+1: GOTO 30
70 CLOSE #1: END
```

Observação: o programa acima lê registros do arquivo criado pelo programa-exemplo apresentado na descrição do comando LOC.

LOF é uma função do DISC BASIC que tem por finalidade registrar a extensão de um arquivo, podendo este ser sequencial ou aleatório.

Sua sintaxe é:

LOF (nº do arquivo)

A extensão do arquivo é registrada em "bytes" ocupados no disquete e é acessível apenas enquanto o arquivo estiver aberto.

Pode ser utilizada na forma direta ou indireta, para diversas finalidades. Exemplos:

```
PRINT LOF(1)
100 IF LOF(2)>340000 THEN PRINT "ESPAÇO NO DISQUETE
    CHEGANDO AO FIM!"
200 R=LOF(1):PRINT "EXTENSÃO DO ARQUIVO EM BYTES:";R
```

Programa-exemplo:

```
10 OPEN "A:LISTEL.ARQ" AS #1 LEN=32
20 FIELD #1,20 AS NM$,12 AS TL$: C=1
30 IF C=10 THEN 100
40 PRINT: INPUT "NOME(máx. 20 carac.):";N$
50 LSET NM$=N$
60 PRINT: INPUT "TEL.(máx. 12 carac.):";T$
70 LSET TL$=T$
80 PUT #1,C: C=C+1
90 GOTO 30
100 PRINT,,"BYTES UTILIZADOS NO ARQ.:";LOF(1)
110 CLOSE #1: END
```

Como vimos na descrição da instrução FIELD, no processamento de arquivos de acesso direto, depois de os dados estarem na memória principal do computador, eles devem ser armazenados transitoriamente num "buffer" antes de passar para a fase de gravação no disquete. Essa é exatamente a função de LSET: transferir dados da memória principal do computador para o "buffer" de transição próprio do sistema de arquivo de acesso direto.

Sua sintaxe é:

LSET variável do campo = variável intermediária ou expressão alfanumérica

Variável do campo é uma variável definida pela instrução FIELD.

Variável intermediária ou expressão alfanumérica é o dado a atribuir à variável do campo, podendo ser uma variável comum, com atribuição feita através de INPUT, LINE INPUT, LET ou READ, ou uma expressão alfanumérica convencional.

O nome da variável intermediária ou comum não pode ser confundido com o da variável definida por FIELD.

Se a expressão alfanumérica, ou a atribuição feita à variável intermediária que a representa, tiver menos caracteres do que os reservados para a variável de campo correspondente, LSET alinha a expressão pela esquerda e justifica, isto é, mantém a área restante com espaços em branco.

Se a expressão alfanumérica tiver mais caracteres do que os espaços reservados para a variável de campo correspondente, serão desprezados os da direita.

Valores numéricos devem ser convertidos para expressões alfanuméricas, por meio das funções MKD\$, MKI\$ e MKS\$, antes de serem transferidos através de LSET para as variáveis de campo definidas por FIELD.

Através da função SPACE\$, a instrução LSET pode ser usada também

para alinhar pela esquerda atribuições de variáveis não definidas por FIELD.

Nos programas listados a seguir são apresentados alguns exemplos do uso de LSET.

Programa-exemplo para criar e gravar dados em arquivo:

```
10 OPEN "FIDESP.ARQ" AS #1 LEN=33
20 FIELD #1,25 AS DES,8 AS VL$: C=1
30 CLS: PRINT "Título da despesa:":PRINT: INPUT D$
40 PRINT:PRINT "Valor da despesa:":PRINT: INPUT D#
50 LSET DES=D$: LSET VL$=MKD$(D#)
60 PUT #1,C: C=C+1
70 PRINT: INPUT "Outro registro (s/n)";R$
80 IF R$="S" OR R$="s" THEN 30 ELSE CLOSE #1: END
```

Programa-exemplo para ler dados do arquivo:

```
200 OPEN "FIDESP.ARQ" AS #1 LEN=33
210 FIELD #1,25 AS DES,8 AS VL$: T=0
220 CLS: FOR C=1 TO LOF(1)/33
230 GET #1,C
240 PRINT DES;TAB(29)CVD(VL$)
250 D=CVD(VL$): T=T+D
260 NEXT C
270 A$=SPACES(28)
280 L$="Total das despesas:  Cz$"
290 LSET A$=L$
300 PRINT A$;USING"#####.###";T
310 CLOSE #1: END
```

A finalidade do comando MAXFILES é especificar o número de arquivos que podem ser abertos pela instrução OPEN num programa.

Sua sintaxe é:

MAXFILES = n₀

onde n₀ é o número de arquivos e pode variar de 1 a 6 para arquivos em disquete.

Quando um programa for operar com um arquivo apenas, não é necessário especificá-lo.

O comando MAXFILES está diretamente ligado à instrução OPEN e determina apenas a quantidade de arquivos que podem ser abertos concomitantemente num programa, para entrada e saída de dados em disquete.

As modalidades de abertura de arquivos e outros detalhes pertinentes são examinados na instrução OPEN.

O comando MAXFILES opera igualmente no BASIC e no DISC BASIC.

A finalidade do comando MERGE é carregar do disquete para a memória do computador um programa em BASIC, previamente gravado no formato ASCII, juntando-o a outro, também em BASIC, já existente nela.

Se for dado o comando MERGE para carregar um programa gravado no formato binário, o processo será abortado e será emitida uma mensagem de erro através do vídeo.

Sua sintaxe é:

```
MERGE "Cd:]nomepro"
```

Se o disquete que contém o programa a ser carregado encontrar-se no DISK DRIVE não corrente, d: deverá ser especificado no comando.

Nomepro deve ser indicado exatamente como foi gravado.

Os programas a serem juntados por este processo devem ter numeração diferente de linhas. Se tiverem números idênticos, as que entrarem através do comando MERGE substituirão as já existentes na memória do computador.

O comando MERGE é muito útil para juntar a programas em processamento rotinas de uso comum e frequentemente adotadas.

Exemplo de comando:

```
MERGE "A:DADAPES.BAS"
```


No processamento de arquivos de acesso direto, os dados que compõem os registros só podem ser arquivados através de variáveis alfanuméricas, que funcionam como divisões internas de uma pasta de cartolina num arquivo convencional de escritório.

Assim, quando for necessário inserir em arquivo de acesso direto números ou variáveis numéricas, é preciso convertê-los para variáveis alfanuméricas ou "strings". Essa é, exatamente, a função desempenhada por MKD\$, MKS\$ e MKI\$.

Sua sintaxe é:

MKD\$(nº de dupla precisão ou
variável correspondente)

MKS\$(nº de simples precisão ou
variável correspondente)

MKI\$(nº inteiro ou variável
correspondente)

Sabemos que no MSX números inteiros podem variar de -32768 até 32767, que números de simples precisão podem ter até 6 dígitos e que números de dupla precisão podem ser representados com até 14 dígitos. Todavia, utilizam respectivamente apenas 2, 4 e 8 "bytes" da memória do computador.

Assim, na instrução FIELD, as variáveis definidas para receber os dados referentes a números - que são as conversões efetuadas por MKD\$, MKS\$ e MKI\$ - são sempre definidas com as quantidades de "bytes" correspondentes aos tipos de números convertidos. Por exemplo:

- se ND\$ for receber os dados referentes a um número de dupla precisão convertido por MKD\$, a definição deverá ser:

FIELD #1,8 AS ND\$

- se NS\$ for assimilar os dados de um número de simples precisão convertido por MKS\$, a definição deverá ser:

FIELD #1,4 AS NS\$

- se NI\$ for armazenar os dados de um número inteiro convertido por MKI\$, a definição deverá ser:

FIELD #1,2 AS NI\$

- se as três modalidades de números forem adotadas para um mesmo arquivo, a instrução deverá ser:

FIELD #1,8 AS ND\$,4 AS NS\$,2 AS NI\$

Os nomes das variáveis poderão ser quaisquer, desde que não sejam idênticos aos de variáveis comuns usadas no mesmo programa, e só poderão receber suas atribuições através de MKD\$, MKS\$ e MKI\$.

As quantidades de "bytes" destinadas às mesmas através da instrução FIELD não poderão ser menores do que as citadas. Se forem superiores, ocorrerá apenas desperdício de espaço no disquete.

RESUMO:

- MKD\$ converte um número de dupla precisão ou uma variável numérica correspondente em variável alfanumérica de 8 "bytes".
- MKS\$ converte um número de simples precisão ou uma variável numérica correspondente em variável alfanumérica de 4 "bytes".
- MKI\$ converte um número inteiro ou uma variável numérica correspondente em variável alfanumérica de 2 "bytes".

MKD\$, MKS\$ e MKI\$ são transferidos da memória principal do computador para o "buffer" de transição próprio do sistema de arquivo aleatório pela instrução LSET ou RSET. Exemplos:

```
RSET DP$=MKD$(27923.014567)
RSET SP$=MK$(29.3456)
RSET IN$=MKI$(1987)
B%=1987: RSET IN$=MKI$(B%)
```

Programa-exemplo:

```
10 OPEN "A:FIPIES.ARQ" AS #1 LEN=14
20 FIELD #1,8 AS DP$,4 AS SP$,2 AS IN$
30 READ A#,B!,C%
40 LSET DP$=MKD$(A#)
50 LSET SP$=MK$(B!)
60 LSET IN$=MKI$(C%)
70 PUT #1,1
80 CLOSE #1
90 END
100 DATA 3.141592,145.612,14
```

Através do comando NAME altera-se ou troca-se o nome de programa existente no disquete.

Sua sintaxe é:

```
NAME "[d:]nomepro antigo" AS  
      "nomepro novo"
```

Nomepro antigo é o nome sob o qual o programa ou arquivo foi gravado no disquete.

Nomepro novo é o novo nome a ser dado ao programa ou arquivo.

Se já existir no disquete programa ou arquivo com nome idêntico ao do novo nome indicado, o processo será cancelado e será emitida no vídeo a mensagem:

```
Arquivo já existe      (Hotbit)
```

```
File already exists    (Expert)
```

A alteração do nome é efetuada sem mudar a localização do programa no disquete.

Se houver dois DISK DRIVES operando no sistema e o programa cujo nome deve ser alterado estiver alojado na unidade não assumida, o dispositivo d: deverá ser especificado.

Exemplos de comandos:

```
NAME "DATAFILE" AS "DADOS.ARQ"  
NAME "BIGFILE" AS "B:BIGDATA.ARQ"
```

A instrução OPEN abre arquivos no disquete, para operações de entrada e saída de dados. Sua sintaxe é:

```
OPEN "[d:]nomearq" FOR modo AS # n°
```

(para arquivo sequencial)

```
OPEN "[d:]nomearq" AS # n° LEN = n°
```

(para arquivo de acesso direto)

Para que possam ser efetuadas operações de entrada e saída (gravação e leitura) de dados no e do disquete, é necessário criar determinadas condições. A primeira delas é abrir o arquivo onde serão armazenados dados para utilização posterior (consulta, relatórios etc.).

Ao ser executada, OPEN reserva na memória do computador uma área específica denominada "buffer" de transição, onde os dados são organizados e armazenados temporariamente antes de serem gravados no disquete e depois de lidos ou recuperados do mesmo.

Nomearq é o nome do arquivo a ser operado para entrada e saída de dados e não deve ser confundido com o nome do programa gerenciador do mesmo. Geralmente, um arquivo é criado e gerenciado por um programa de nome diferente do arquivo de dados correspondente.

Modo é condição válida apenas para arquivos sequenciais e significa a modalidade de operação para a qual o arquivo é aberto, podendo ser:

OUTPUT para saída de dados do computador para o disquete.

INPUT para entrada no computador de dados lidos no disquete.

APPEND para acréscimo de registros num arquivo sequencial já formado.

n° significa o número do arquivo em processamento, sendo associa-

do a ele apenas enquanto aberto. Pode variar de 1 até o limite estabelecido pela instrução MAXFILES, não podendo exceder 6 para arquivos em disquete.

nº' significa a quantidade de "bytes" para cada registro do arquivo, a ser utilizada de acordo com a instrução FIELD. Se não for especificada, será assumida automaticamente pelo sistema a máxima, que é de 256 "bytes", o que representará desperdício de espaço no disquete se a quantidade efetivamente necessária para cada registro for menor.

Depreende-se das sintaxes apresentadas que não é preciso indicar o modo de operação na abertura de arquivos de acesso direto, havendo conveniência, todavia, em especificar a extensão de cada registro em "bytes", para evitar desperdício de espaço no disquete.

Podem ser abertos concomitantemente até seis arquivos de dados em ambos os tipos, sequencial e de acesso direto, para entrada de dados. Para saída de dados, porém, pode ser aberto apenas um arquivo por vez.

Exemplos de comandos:

```
OPEN "A:FIPES.ARQ" FOR OUTPUT AS #1
```

```
OPEN "FIDESP.ARQ" AS #1 LEN=33
```


PRINT # e PRINT # USING DISC BASIC

A instrução PRINT # é utilizada para gravar ou arquivar dados de arquivo sequencial em disquete.

Assim como PRINT (sem #) "imprime" no vídeo dados ou expressões, PRINT # imprime-os magneticamente no disquete, isto é, grava-os.

Sua sintaxe é:

```
PRINT # no, lista de expressões  
ou de variáveis
```

```
PRINT # no USING "formato"; lista  
de expressões ou de variáveis
```

n_o é o número do arquivo aberto pela instrução OPEN para inserção de dados, isto é, no modo FOR OUTPUT.

Formato é a formatação que pode ser adotada para arquivamento dos dados, conforme recursos da instrução PRINT USING do MSX BASIC.

Lista de expressões ou de variáveis são os dados numéricos ou alfanuméricos a inserir no arquivo, na sua forma natural ou representados por variáveis.

Na forma natural, os dados são arquivados de maneira direta, como nos exemplos que seguem:

```
PRINT #1,1987
```

```
PRINT #1,"ANO DA CONSTITUINTE"
```

Representados por variáveis, os dados são arquivados de forma indireta, como nos exemplos que seguem:

```
A=1987:A$="ANO DA CONSTITUINTE"
```

```
PRINT #1,A:PRINT #1,A$
```

Para que a instrução PRINT # seja válida é preciso existir um arquivo aberto pela instrução OPEN.

Para que os dados possam ser lidos ou recuperados adequadamente, certos cuidados devem ser tomados ao gravá-los no disquete. Tais cuidados referem-se especificamente às delimitações que devem receber, a fim de serem tratados pelo sistema operacional de forma conveniente.

Expressões ou variáveis numéricas devem ser separadas preferivelmente por ponto-e-vírgula, em vez de apenas por vírgulas. Exemplo:

```
PRINT #1,A;B;N;Y
```

ou

```
PRINT #1,15;11;1987
```

Expressões ou variáveis alfanuméricas ("strings") devem ser separadas por ponto-e-vírgula, vírgula entre aspas e, novamente, ponto-e-vírgula (;",","). Exemplo:

```
PRINT #1,C$;"",";N$
```

ou

```
PRINT #1,"ANO";",","NOVO"
```

Se a própria expressão alfanumérica - como uma frase, por exemplo - contiver vírgulas, pontos-e-vírgulas, vários espaços em branco, mas não aspas internas, mas sim apenas aspas delimitadoras, deverá ser arquivada no disquete circunscrita entre aspas explicitadas pela função CHR\$(34) com o código correspondente entre parênteses - CHR\$(34) - , sendo todos separados por ponto-e-vírgula. Exemplo num pequeno programa:

```
10 A$ = "SANGUE, SUOR, LAGRIMAS"
20 B$ = "PENSOU; DISSE 'NÃO!'. FOI PRESO."
30 OPEN "TESTE" FOR OUTPUT AS #1
40 PRINT #1,CHR$(34);A$;CHR$(34);CHR$(34);B$;CHR$(34)
50 CLOSE #1: END
```

Tal situação poderá também ser contornada através da repetição do comando PRINT # para cada variável ou expressão alfanumérica. Exemplo:

```
PRINT #1, A$:PRINT #1,B$
```

devendo o programa relativo de leitura usar a instrução LINE INPUT # em vez de INPUT #, na forma exemplificada a seguir:

```
LINEINPUT #1,A$:PRINT A$
```

```
LINEINPUT #1,B$:PRINT B$
```

Observação: quando a atribuição de uma variável alfanumérica a ser arquivada através de PRINT # puder ser feita pela instrução LINE INPUT, as exigências descritas são contornadas. Por exemplo, a frase da linha 20 do pequeno programa apresentado poderá ser introduzida através de uma instrução como:

```
20 LINE INPUT B$
```

sendo digitada durante a execução do programa - em tal caso inclusive com aspas em "NÃO!" - e gravada simplesmente pela instrução PRINT # na sua forma mais simples:

```
40 PRINT #1,B$
```

É claro que a mesma solução se aplica também à linha 10.

Em tal caso, a leitura correspondente dos registros do arquivo só poderá ser feita através da instrução LINE INPUT #, obviamente.

PUT é um comando exclusivo do processamento de arquivos de acesso direto ou aleatório.

Sua função é acionar os recursos do sistema operacional para que seja gravado no disquete um registro pronto, isto é, um registro que já recebeu dados para todos os seus campos, mesmo que sejam "dados em branco".

Sua sintaxe é:

PUT #n₀,n₀'

n₀ é o número adotado na instrução OPEN para abertura do arquivo.

n₀' é o número do registro que deverá ser gravado. Se for omitido, será assumido automaticamente o número subsequente à última execução de um comando PUT.

Os números de registros podem variar de 1 a 4.294.967.295 e são estabelecidos por intermédio de uma variável contadora qualquer.

A instrução PUT só é executável após um arquivo ter sido aberto pela instrução OPEN, os campos do registro terem sido estabelecidos por FIELD e os dados respectivos terem sido colocados no "buffer" próprio do sistema por um dos comandos LSET ou RSET.

Programa-exemplo:

```
10 OPEN "A:DATATEL.ARQ" AS #1 LEN=32
20 FIELD #1,20 AS N$,12 AS T$: C=1
30 INPUT "NOME:";X$: IF X$="?" THEN 90
40 LSET N$=X$
50 INPUT "TEL.:";X$
60 RSET T$=X$
70 PUT #1,C: X$=""
80 C=C+1: GOTO 30
90 CLOSE #1: END
```

Desempenha exatamente a mesma função de LSET, com a única diferença de que alinha as palavras pelo lado direito, ao passá-las da memória principal do computador para as variáveis de campo definidas por FIELD.

Os programas apresentados a seguir ilustram o uso de RSET.

Programa-exemplo para criar e gravar dados em arquivo:

```
10 OPEN "cardap.arq" AS #1 LEN=30
20 FIELD #1,26 AS PR$,4 AS CZ$
30 FOR C=1 TO 5
40 READ R$,R!
50 LSET PR$=R$
60 RSET CZ$=MK$$(R!)
70 PUT #1,C: NEXT: CLOSE
80 DATA PRATO DO DIA      CZ$,30.00
90 DATA TUTU DE FEIJÃO    CZ$,55.00
100 DATA PESCADA A BAIANA  CZ$,75.00
110 DATA CAMARÃO A GREGA  CZ$,120.00
120 DATA TORTA DE MAÇA    CZ$,20.00
130 END
```

Programa-exemplo para ler dados do arquivo:

```
200 CLS: PRINT, ,TAB(9)"RESTAURANTE JABURU": PRINT
210 OPEN "cardap.arq" AS #1 LEN=30
220 FIELD #1,26 AS PR$,4 AS CZ$
230 FOR C=1 TO 5
240 GET #1,C: PRINT
250 PRINT, ,PR$;TAB(30)USING"###.##";CVS(CZ$)
260 NEXT: CLOSE
270 END
```

Este comando tem por finalidade carregar um programa na memória do computador e iniciar sua execução.

Sua sintaxe é:

RUN "[d:]nomepro"[,R]

A indicação de d: deve ser feita no comando se o disquete em que se encontra o programa a ser carregado estiver no DISK DRIVE não assumido.

Ao ser dado o comando RUN, todas as variáveis definidas e vigentes na memória do computador são canceladas e arquivos eventualmente abertos são fechados.

OPÇÃO R - Evita o fechamento de arquivos que se encontram abertos no computador no momento de execução do comando.

Se se desejar executar o programa a partir de linha de instrução que não seja a inicial, deverá ser indicado seu número no comando, na forma exemplificada a seguir:

RUN número de linha

Exemplos de comandos:

RUN "A:FIDESP.BAS"

RUN "CADAPES",R

RUN 700

Através do comando SAVE, um programa existente na memória do computador é gravado ou armazenado em disquete. Sua sintaxe é:

```
SAVE "[d;]nomepro"[A]
```

A título de experiência, digite o pequeno programa que segue:

```
10 PRINT "COMANDO SAVE"  
20 PRINT "GRAVA PROGRAMAS NO DISQUETE"  
30 END
```

e, havendo no DISK DRIVE corrente um disquete, dê o seguinte comando:

```
SAVE "PROGRAM1"
```

O DISK DRIVE será acionado automaticamente e aparecerá em seguida a palavra OK no vídeo, com o cursor logo abaixo, indicando a finalização do processo. O pequeno programa estará gravado no disquete.

Como vimos antes, no DISC BASIC é possível usar até oito caracteres no nome do programa. E pode também ser adotada uma extensão para o nome, com até três letras, sempre precedida por um ponto. Exemplo:

```
PROGRAM1.BAS
```

A finalidade da extensão é permitir melhor identificação e classificação do programa, já que, em vista de ser possível o armazenamento de elevado número deles num disquete, é necessário dotá-los de elementos que permitam selecionar ou identificar mais rapidamente um programa especificamente desejado, ao ser examinada a lista de seus nomes quando esta é fornecida por um dos comandos "files" ou "lfiles".

Quaisquer palavras com até oito caracteres para o nome do programa e com até três letras para a extensão do mesmo podem ser usadas na denominação de programas ou arquivos, desde que não sejam as especiais reservadas ao sistema e citadas a seguir:

AUX, CON, LST, PRN e NUL.

Seguem alguns exemplos de extensões normalmente adotadas para nomes de programas ou de arquivos e sua significação:

- .TXT - arquivo de texto.
- .BAS - programa em BASIC no formato binário.
- .ASC - programa em BASIC no formato ASCII.
- .ARQ - arquivo em geral.
- .LM - programa em linguagem de máquina.
- .ASM - programa em "assembler".
- .SYS - programa próprio do sistema operacional.
- .COM - programa de comandos do sistema operacional.
- .BAT - comando coletivo auto-executante.

Quando houver apenas um DISK DRIVE conectado no computador, não é necessário especificar d: no comando SAVE.

Se houver dois DISK DRIVES operando no sistema e o disquete a receber a gravação estiver no DISK DRIVE assumido, seja A ou B, também não há necessidade de indicar d: no comando SAVE.

Se, porém, o disquete a receber a gravação estiver no DISK DRIVE não assumido, este deverá ser indicado no comando. Supondo que seja o caso do DISK DRIVE B, o comando deverá ser:

SAVE "B:PROGRAM1.BAS"

Note que DISK DRIVE deve ser indicado sempre como A: ou B: (a letra seguida de dois-pontos) e que o nome do programa e a sua extensão devem estar separados por um ponto (.), sendo o conjunto colocado entre aspas.

Outro ponto importante a notar: se for adotado para o programa a ser gravado um nome de programa já existente no disquete, o novo será sobreposto ao antigo, apagando-o. Não é emitida nenhuma mensagem pelo

sistema operacional, alertando o operador que já existe programa com nome idêntico no disquete.

Por tal razão, é conveniente sempre consultar o diretório do disquete (comando FILES) antes de se dar um nome ao programa a ser gravado, principalmente quando não se mantêm habitualmente cópias ou duplicatas dos programas.

Na forma acima descrita, as gravações de programas são executadas pelo comando SAVE em formato binário, correspondente ao comando CSAVE do MSX BASIC para gravações em fita cassete.

OPÇÃO A - Para gravar programas em disquete no formato ASCII, é necessário acrescentar no final do comando de gravação a letra A precedida de uma vírgula. Exemplo:

```
SAVE "PROGRAM1.BAS",A
```

Um programa em disquete gravado no formato ASCII pode ser juntado a outro existente na memória do computador, através do comando MERGE, da mesma forma que no MSX BASIC.

É oportuno lembrar que a gravação de programas no formato ASCII utiliza mais espaço no disquete do que a de formato binário. Por exemplo, um mesmo programa que utilize

6280 "bytes" no formato binário, consumirá

8607 "bytes" no formato ASCII.

A gravação de formato ASCII também exige mais tempo do que a de formato binário. Por exemplo, um programa que demora dois segundos para ser gravado no formato binário demorará aproximadamente 12-14 segundos para ser gravado no formato ASCII. Detalhes sobre tal assunto são encontrados no capítulo "GRAVAÇÕES NOS FORMATOS BINÁRIO E ASCII".

O comando SAVE do DISC BASIC não deve ser confundido com o mesmo comando do BASIC. No BASIC, o comando deve especificar obrigatoriamente o dispositivo de gravação como CAS: (gravador cassete) e executa a gravação sempre no formato ASCII.

O comando SYSTEM permite retornar ao sistema MSX DOS, depois de ter sido o controle operacional passado para o BASIC e, por extensão, ao DISC BASIC. Sua sintaxe é:

CALL SYSTEM ou _SYSTEM

O comando é válido apenas quando o sistema operacional para disco foi inicializado com o MSX DOS, tendo sido o controle transferido para o BASIC através do comando de mesmo nome, próprio do sistema DOS.

Como vimos antes, estando ativado o sistema DOS, não é possível operar o computador em BASIC.

Assim, para se digitar ou carregar um programa na memória do computador, executá-lo ou gravá-lo, ou operar o computador normalmente, é preciso passar o controle do mesmo para o BASIC.

Por outro lado, estando o computador nessa modalidade - portanto, com o DISC BASIC ativado - o processamento de disquetes pode ser feito da mesma forma, já que, além de comandos praticamente equivalentes aos do MSX DOS, o DISC BASIC possui muitos outros, aliados a recursos sobejamente suficientes para a finalidade.

Para que o comando SYSTEM se efetive, é necessário também que esteja alojado no DISK DRIVE operante um disquete que contenha os dois programas que constituem o MSX DOS, pois, como vimos, esse sistema reside exclusivamente em disquete.

Outro ponto importante a ser observado é que, ao ser executado o comando SYSTEM, todo o conteúdo da RAM é apagado, em razão de ocorrer mudança nas áreas de memória do computador na transferência de um para outro sistema operacional.

VARPTR é uma função do sistema operacional que fornece o endereço inicial do bloco de controle do "buffer" do arquivo especificado.

Sua sintaxe é:

VARPTR (# n₀)

n₀ é o número do arquivo aberto pela instrução OPEN.

Se o endereço fornecido por VARPTR for um número negativo, deverá ser somado a 65536 para se obter o endereço real.

Exemplo:

```
W=VARPTR(#1):PRINT W
```

O programa apresentado a seguir obtém o endereço inicial do bloco de controle de um "buffer" de arquivo. Utilizando o 2^o e 3^o "bytes" desse bloco, obtém o endereço inicial do BLOCO DE CONTROLE DO ARQUIVO. Em seguida, coleta os dados registrados nesse bloco e os exibe na forma de códigos hexadecimais para leitura e interpretação.

A execução do programa só é válida após um arquivo ter sido aberto pela instrução OPEN.

```
5000 W=VARPTR(# 1)
5010 P=PEEK(W+1)+PEEK(W+2)*256
5020 FOR C=P TO P+36
5030 V$=HEX$(PEEK(C)):V$=RIGHT$("00"+V$,2)
5040 PRINT V$;" ";
5050 NEXT C: END
```

No Apêndice existente no final do livro encontra-se um mapa com o formato de um BLOCO DE CONTROLE DE ARQUIVO e o significado dos códigos contidos no mesmo.

O comando VERIFY liga ou desliga o "modo verificação" para gravações de programas em disquete.

Sua sintaxe é:

CALL VERIFY ON ou OFF

ou _VERIFY ON ou OFF

No modo VERIFY ON, a gravação executada no disquete é verificada, isto é, conferida com o programa existente na memória do computador, à medida em que é efetuada.

Se for constatado pelo sistema algum erro, o processo de gravação é interrompido e a ocorrência é indicada através da seguinte mensagem exibida no vídeo:

ERRO I/O ou DISK I/O ERROR

devendo a gravação ser refeita.

VERIFY ON ativa o modo verificação e VERIFY OFF desativa. O modo assumido espontaneamente ao ser o sistema operacional ativado, ou modo "default", é o VERIFY OFF.

O processo de gravação de programas em disquete com o modo verificação ligado demanda mais tempo do que o processo normal, mas oferece absoluta confiabilidade quanto ao resultado.

9.3 - PROGRAMA AUTO-EXECUTANTE - AUTOEXEC.BAS

Sempre que o DISC BASIC é ativado estando alojado no DISK DRIVE corrente um disquete que não contém os programas que compõem o MSX DOS ou o HB-MCP, o sistema operacional procura espontaneamente no disquete um programa com o nome:

AUTOEXEC.BAS

e, se ele existir no disquete, é automaticamente carregado para a memória do computador e executado.

Frisamos: tal ação do DISC BASIC só ocorre quando o computador é ligado ou "resetado", estando alojado no DISK DRIVE assumido o disquete que contém um programa com a denominação AUTOEXEC.BAS e que não estejam gravados no mesmo disquete os programas que compõem um dos sistemas MSX DOS e HB-MCP.

Qualquer programa em BASIC que seja gravado com o nome AUTOEXEC.BAS se enquadra nessa condição, ou seja: é auto-executante.

Num mesmo disquete pode haver apenas um programa auto-executante.

Programa "fechado" ou "protegido", que não permite interrupção ou retorno aos comandos do BASIC, não deve ser utilizado com o recurso do AUTOEXEC.BAS, pois, para interromper sua execução, é preciso desligar o computador e, ao ser restabelecido seu funcionamento, o programa será novamente carregado na memória do computador e executado automaticamente, se o disquete que o contiver continuar alojado do DISK DRIVE, fazendo com que tal disquete só possa ser utilizado por esse único e mesmo programa.

Se lembrarmos que um disquete de 5,25 polegadas utilizado em computador padrão MSX pode armazenar 354 "kbytes", um programa que o utilize com exclusividade poderá representar injustificável desperdício, a não ser que seja um programa gerenciador de arquivo de dados, com o respectivo arquivo, e que demande tal capacidade de armazenamento.

Algumas utilizações recomendáveis para o recurso AUTOEXEC.BAS:

- exibição automática do diretório do disquete. Para tanto,

bastará gravar um programa de uma linha, como a que segue:

10 FILES

e gravá-lo no disquete com o nome AUTOEXEC.BAS;

- exibição automática de um "menu" dos programas utilizados com maior frequência e com recursos para execução automática do escolhido mediante a pressão de uma única tecla;
- carregamento e execução automática de um programa gerenciador de banco de dados ou de arquivo de texto.

9.4 - OCORRÊNCIAS DE ERROS EM OPERAÇÕES DO MSX DISC BASIC E MENSAGENS DO SISTEMA

Quando ocorrem erros em operações do DISC BASIC, seu sistema operacional emite mensagens, através do vídeo, reportando a ocorrência, a fim de orientar o usuário ou operador.

Geralmente tais erros se originam de comandos dados incorretamente, nomes de programas ou arquivos digitados com falhas, disquetes não formatados ou sobrecarregados com gravações, erros na execução de programas etc., sendo sanáveis na quase totalidade.

Em raros casos, o erro é oriundo do próprio disquete - denominado "erro de disco" -, por falha material ou de formatação, sendo na maioria das vezes irrecuperável.

Na "TABELA DE MENSAGENS DO DISC BASIC" encontram-se todas as mensagens emitidas pelo sistema operacional e seus significados.

10 - COMANDOS E RECURSOS DO MSX DOS

10.1 - TABELA DOS COMANDOS DO MSX DOS

BASIC	DIR	REM
COPY	FORMAT	REN - RENAME
DATE	MODE	TIME
DEL - ERASE	PAUSE	TYPE
		VERIFY

10.2 - DESCRIÇÃO DOS COMANDOS DO MSX DOS

Para facilitar consultas, os comandos integrantes do MSX DOS serão apresentados em ordem alfabética nas páginas seguintes, começando sempre no topo de cada uma, com os nomes em caracteres destacados.

Após os comandos serão apresentados outros recursos do MSX DOS.

Estando o computador operando com o MSX DOS, o comando BASIC permite que se passe a operá-lo com o BASIC e o DISC BASIC.

Sua sintaxe é:

BASIC [nomepro]

Se, ao ser dado o comando, for especificado após o mesmo o nome de um programa em BASIC existente no disquete, o controle do computador passará para o BASIC e o programa será automaticamente carregado para a memória do computador e executado.

Nessa condição, também o DISC BASIC poderá ser operado normalmente, permitindo a qualquer momento retornar ao MSX DOS, o que é efetuado através do comando CALL SYSTEM.

Ocorre que, sendo o mapeamento de memória diferente entre o DISC BASIC e o MSX DOS, quando é feito o retorno a este, qualquer programa que se encontre na memória do computador será destruído, bem como variáveis definidas serão anuladas e arquivos eventualmente abertos serão fechados.

Suponha-se que, estando operando em BASIC e havendo na memória do computador um programa em digitação, pretenda-se retornar ao MSX DOS por instantes, para, por exemplo, verificar por intermédio do comando TYPE alguns dados de um arquivo em disquete que devem ser utilizados no programa em digitação. Se, em tal situação, for dado o comando CALL SYSTEM, tudo que estiver na memória do computador será apagado e todo o trabalho já feito será perdido, a menos que, antes do retorno ao MSX DOS, o programa existente na RAM seja salvo normalmente através do comando SAVE do DISC BASIC.

Sem dúvida, tal providência vale e deve ser tomada, já que, para gravar o programa em digitação, retornar ao MSX DOS, verificar através de TYPE o que se deseja, invocar outra vez o BASIC e carregar novamente na memória do computador o programa que está sendo trabalhado, não serão despendidos mais do que alguns segundos!

A função do comando COPY é fazer o sistema operacional copiar arquivos ou programas de um disquete para outro.

Sua sintaxe é:

COPY nomepro [d:] nomepro [d:][A][B][V]

Omitindo-se a indicação do dispositivo, um programa pode ser copiado no mesmo disquete com nome diferente. Exemplo:

COPY NEWDATA.BAS NOVADATA.BAS

A duplicação de um mesmo programa num disquete pode ser bem útil quando se deseja fazer alterações num programa e estudar seus efeitos, mantendo-se o original intacto.

Para copiar um programa ou arquivo de um disquete para outro, sem alterar seu nome, o comando a ser dado é o que segue:

COPY nomepro B:

quando o mesmo se encontra no disquete alojado no DISK DRIVE A.

Se houver apenas um DISK DRIVE operando no sistema, os comandos são dados na mesma forma de quando há duas unidades em operação. (Consulte o Capítulo 7, "OPERANDO COM UM OU DOIS DISK DRIVES".)

Para copiar mais de um programa por comando, podem ser utilizados os caracteres de função especial ? e *. Por exemplo:

COPY *.BAS B:

copiará de um disquete para outro todos os programas cuja extensão do nome seja .BAS.

COPY HIPOT?.* B:

copiará de um disquete para outro todos os programas cujos nomes sejam iguais a HIPOT nos cinco primeiros caracteres, sendo o sexto caractere e a extensão quaisquer.

Podem ser usados vários sinais de interrogação no nome do programa ou em sua extensão. Exemplos:

COPY TXT???.ARQ B:

COPY CIR?N? B:

Para copiar todos os programas de um disquete em outro, sob um só comando, é usado o caractere especial * tanto em lugar do nome como no da extensão. Exemplo:

COPY *.* B:

O comando COPY pode ser usado também para CONCATENAR ou JUNTAR programas. Por exemplo, através do comando

COPY FIPES.BAS+FLS.TXT RESUM.BAS

são concatenados ou juntados os programas FIPES.BAS e FLS.TXT num terceiro programa denominado RESUM.BAS.

Quando usado no processo de cópia-concatenação o caractere especial * em lugar de nomes, não deve ser dado ao arquivo final nome que seja igual ao de um programa ou arquivo em concatenação, já que o mesmo será destruído ao ser detectada a coincidência pelo sistema. Exemplifica-se a seguir um comando que costuma causar tal problema:

COPY *.TXT MEMO.TXT

Durante o processo de concatenação, o sistema operacional compara o nome do arquivo a ser concatenado com o do arquivo final. Se houver no disquete um programa ou arquivo com o mesmo nome, ele será destruído sem interrupção do processo, sendo exibida no vídeo a mensagem:

Conteúdo perdeu-se antes da cópia

prossequindo normalmente.

O problema será contornado se, na concatenação, for usado um programa já existente, ao qual se juntem outros. Exemplo:

COPY FIPES.ARQ+*.ARQ

sendo este, por sinal, o melhor modo de efetuar concatenações.

COPY pode também fazer a impressora reproduzir - não se trata de listar - programas em BASIC gravados no formato ASCII e registros de arquivos de dados. Comandos para esse fim são exemplificados a seguir:

COPY FIPES.ARQ PRN

COPY MALADIR.ASC PRN

Deve ser observado, todavia, que os caracteres enviados à impressora são apenas os da tabela ASCII, portanto sem a acentuação gráfica da língua portuguesa e "c" com cedilha. Tal fato ocorre porque o sistema operacional DOS é originário de país de língua inglesa, onde as palavras não são acentuadas graficamente.

No processo de cópia-concatenação, o comando COPY pode ser complementado por chaves ou opções como /A, /B e /V.

CHAVE A/ - é usada na concatenação de arquivos ou programas gravados no formato ASCII. Remove dos arquivos em concatenação os códigos de fim de arquivo (&H1A) e adiciona ao arquivo final o mesmo código. É sobretudo indispensável na concatenação de arquivos sequenciais.

CHAVE /B - é usada na concatenação de arquivos (formato ASCII) e/ou programas em formato binário a fim de que os códigos &H1A sejam ignorados mas não removidos durante a concatenação. Não adiciona automaticamente um código &H1A ao arquivo final, sendo necessário incluir no nome deste a chave /A se tal código dever constar dele. É deveras útil na junção de arquivos de acesso aleatório, a fim de não alterar o LOF.

CHAVE /V - ativa a modalidade de verificação durante o processo de cópia-concatenação (veja o comando VERIFY).

Por exemplo, na concatenação:

COPY ARTE1/B+ARTE2/A

a chave /B evita remover o código &H1A do primeiro arquivo e a chave /A adiciona o mesmo código ao segundo arquivo.

Assim, para que seja adicionado no final da concatenação um código &H1A válido para o arquivo-destino, é usada neste a chave /A:

COPY ARQ1+ARQ2+ARQ3 ARQFINAL/A

caso em que os arquivos ARQ1, ARQ2 e ARQ3 são juntados em ARQFINAL e este recebe o código de fim de arquivo para processamento normal.

A chave /V pode ser usada em conjunto com as chaves /A e /B, sendo aconselhável sua utilização quando se deseja conferir maior confiabilidade ao processo. Exemplo:

COPY ARQ1/A+ARQ2/A ARQT/A/V

Obrigatoriamente, DATE é o primeiro comando do MSX DOS com o qual se tem contato quando o sistema é ativado.

Como vimos no Capítulo 6, ao ser ativado o MSX DOS, são exibidos no vídeo alguns dados referentes ao sistema e a data corrente, isto é, a constante no diretório do disquete utilizado, sendo o usuário induzido a entrar com nova data, através da seguinte mensagem:

Entre c/nova data:

ou **Enter new date:**

A data apresentada poderá ser mantida, caso em que bastará teclar RETURN, ou poderá ser alterada. Neste caso, deverá ser digitada a nova data na forma

dia-mês-ano

ou **mês-dia-ano**

conforme for aceito pelo sistema.

Além do hífen (-), podem ser usados o ponto (.) e a barra (/) para separar os dados da data.

Dia e mês devem ser digitados sempre dentro de seus limites válidos, isto é, dia entre 1 e 31, mês entre 1 e 12.

O ano pode ser digitado com dois ou quatro dígitos.

Se for dada entrada a uma nova data, esta substituirá a anterior no diretório do disquete e será a data corrente no mesmo até que seja alterada.

Todos os programas gravados ou regravados - quando alterados - no disquete são automaticamente cadastrados no seu diretório com a data corrente no mesmo, para fins de controle do usuário.

Assim, quando um comando DIR é executado, os nomes dos programas existentes no disquete são listados no vídeo com a data vigente no diretório no momento em que foram gravados, razão por que é aconselhável atualizá-la sempre que possível.

A data corrente no disquete poderá ser verificada e/ou alterada sempre que o cursor próprio do MSX DOS (>) estiver em prontidão de comando. Bastará digitar

DATE

e teclar RETURN e serão apresentados no vídeo os seguintes dados:

Data corrente: Seg 14/07/1986

Entre c/nova data:

Current date is Mon 07-14-1986

Enter new date:

devendo ser adotado o procedimento que convier ao operador, em conformidade com o que foi descrito no início deste item.

A finalidade do comando DEL é deletar ou apagar no disquete programas ou arquivos.

Sua sintaxe é:

DEL ou ERASE [d:]nomepro.

O DISK DRIVE (d:) deverá ser especificado se o sistema operacional tiver duas unidades acopladas e o disquete a ser processado se encontrar na unidade não assumida.

O comando DEL pode eliminar do disquete um programa ou arquivo a cada vez ou todos sob um único comando. Se, em vez do nome de um programa ou arquivo, for feita a indicação exemplificada a seguir:

DEL *.*

todos os programas e arquivos de um disquete serão eliminados. Antes, porém, de executar o comando, o sistema operacional emitirá a seguinte mensagem-pergunta através do vídeo:

Tem certeza (S/N)?

Are you sure (Y/N)?

e procederá de acordo com a resposta que for digitada. Por exemplo, se a resposta for N (=não), o comando será abortado e nenhum programa será apagado no disquete.

Se, porém, a resposta digitada for Y ou S (=sim), o usuário terá em poucos segundos um disquete desocupado e formatado, pronto para receber novos programas e arquivos.

Os caracteres especiais ou curingas - ? e * - também poderão ser usados com o comando DEL. A combinação dos dois caracteres especiais permite deletar seletivamente vários programas ou arquivos sob um úni-

co comando. Por exemplo, o comando:

DEL FICHA??.*

eliminará do disquete todos os programas ou arquivos que possuam como primeiras cinco letras do nome a palavra FICHA, sendo quaisquer a sexta e a sétima letras e a extensão do nome.

DEL *.ARQ

deletará do disquete processado pelo comando todos os programas ou arquivos que tenham no nome a extensão .ARQ.

As palavras DEL e ERASE poderão ser usadas indistintamente para a mesma finalidade.

A finalidade do comando DIR é listar no vídeo o(s) nome(s) de arquivo(s) ou programa(s) existente(s) num disquete.

Sua sintaxe é:

DIR [d:][nomepro][L/P][W]

Se houver dois DISK DRIVES operando com o computador e o disquete a receber o comando estiver na unidade não assumida, d: deverá ser especificado.

Se for especificado nomepro e este constar do diretório do disco, será exibido na tela de vídeo com a extensão em "bytes" e data de gravação respectiva.

Se não for especificado nomepro, serão listados no vídeo todos os nomes de programas e arquivos existentes no disquete, com as respectivas extensões em "bytes" e datas de gravação.

No final da lista é informada a disponibilidade de espaço no disquete, em "bytes".

Quando houver num disquete quantidade elevada de programas - pelo menos mais do que 23 - pode ser usada a

OPÇÃO /P

que interrompe a exibição da lista de nomes quando a mesma toma toda a tela de vídeo - o que equivale a 23 linhas ou nomes de programas -, de forma que o operador possa examiná-la adequadamente.

Pressionando-se em seguida qualquer tecla, a listagem tem prosseguimento nas mesmas condições.

Se for usada a

OPÇÃO /W

serão apresentados no vídeo apenas os nomes dos programas, sem as respectivas extensões em "bytes" e datas de gravação.

Com a OPÇÃO /W os nomes dos programas são apresentados em sequência horizontal, isto é, um após outro em cada linha, de forma a listar o maior número possível deles em cada tela de vídeo. Lembremos que cabem 112 nomes de programas ou arquivos no diretório de um disquete de FACE DUPLA/DENSIDADE DUPLA e 64 no de FACE SIMPLES/DENSIDADE SIMPLES.

Os caracteres especiais * e ? podem ser usados com o comando DIR para seleccionar grupo(s) de programas ou arquivos.

Seguem exemplos de alguns comandos com outros equivalentes:

DIR	=	DIR *.*
DIR nomepro	=	DIR nomepro.*
DIR .EXT	=	DIR *.EXT
DIR .	=	DIR *.

Sabemos já que um disquete novo não pode ser usado sem estar preparado ou "formatado" especificamente para o sistema operacional para disco em que vai ser usado.

A finalidade do comando FORMAT é exatamente a de conferir ao disquete as características próprias do sistema em que vai ser utilizado.

Sua sintaxe é simplesmente o seu próprio nome:

FORMAT

O resultado da execução do comando FORMAT do MSX DOS é o mesmo do comando equivalente do DISC BASIC, pela razão de o programa responsável pela formatação dos disquetes residir na INTERFACE do sistema, o que explica também a perfeita compatibilidade no processamento de programas e arquivos em disquetes entre os dois sistemas.

Como vimos no Capítulo 3, a INTERFACE própria para computadores do padrão MSX é programada para formatar disquetes de 5,25 e 3,5 polegadas, o que possibilita a adoção de DISK DRIVES de qualquer das duas medidas para uso com os mesmos.

Ao ser dado o comando FORMAT, o operador será induzido, por meio de mensagens exibidas no vídeo, a informar ao sistema operacional, via teclado:

- qual o DISK DRIVE a ser acionado para a operação.
- qual a medida do DISK DRIVE.
- se a formatação deve ser feita em uma ou nas duas faces do disquete e se as trilhas devem ser simples ou duplas.

Dependendo da procedência da INTERFACE, tais mensagens podem variar quanto à forma, sendo emitidas em português ou em inglês. O seu conteúdo, todavia, é o mesmo.

As informações a serem transmitidas ao sistema operacional, atra-

vés do teclado, devem ser baseadas no fato de que a formatação a ser conferida ao disquete deve ser compatível com suas características e as do DISK DRIVE adotado para o sistema, não sendo possível utilizar um disquete de FACE DUPLA/DENSIDADE DUPLA se o DISK DRIVE é produzido para processar discos de FACE SIMPLES/DENSIDADE SIMPLES, por exemplo.

No Capítulo 5 são encontrados detalhes sobre os DISK DRIVES que podem ser utilizados com os computadores do padrão MSX, bem como sobre as formatações dos disquetes que podem ser usados neles.

Ao terminar a formatação do disquete, o sistema operacional emite a seguinte mensagem através do vídeo:

Formatação completa

ou Format complete

CARACTERÍSTICAS DE FORMATAÇÃO DE DISQUETES

DOS SISTEMAS MSX DOS E DISC BASIC

	FACE DUPLA	FACE SIMPLES
BYTES POR SETOR	512	512
SETORES POR GRUPO	2	1
SETORES RESERVADOS	1	1
TABELAS DE ALOCAÇÃO	2	2
ENTRADAS NO DIRETÓRIO	112	64
SETORES NO DISQUETE	720	360
SETORES POR TABELA	2	2
SETORES POR TRILHA	9	9
TOTAL EM BYTES	368.640	184.320

Através do comando MODE é determinada a largura da tela de vídeo, ou seja: é especificada a quantidade de colunas da tela de vídeo que serão utilizadas pelo sistema operacional para exibição de dados referentes aos programas ou arquivos processados.

Sua sintaxe é:

MODE n_o

sendo que n_o representa um número que poderá variar até 40 ou, havendo no computador cartucho para expansão de colunas, até 80.

Se o número especificado for igual ou menor do que 32, o modo de operação de vídeo SCREEN 1 é selecionado automaticamente.

MODE é especialmente útil quando é acionado o comando TYPE para verificação de arquivos de texto, com dados gravados em variáveis cuja extensão em "bytes" é regular.

Outra utilização bastante prática de MODE é quando a tela do monitor ou de TV expande seu conteúdo além dos cantos e margens, tornando sua visualização incômoda.

No computador HOTBIT, a largura de tela de vídeo assumida automaticamente quando o mesmo é ligado é de 39 colunas.

No computador EXPERT é de 40.

No Japão esse padrão é 29.

Nos Estados Unidos da América é 39.

Na Inglaterra, Dinamarca, França e outros países é 37.

A finalidade do comando PAUSE é suspender a execução de um COMANDO COLETIVO ou processamento em lote.

Sua sintaxe é:

PAUSE [comentários]

Comentários são geralmente inseridos num COMANDO COLETIVO, a fim de lembrar o operador sobre alguma providência a ser tomada durante a sua execução, como, por exemplo, trocar o disquete no DISK DRIVE operante, ligar a impressora etc.

Quando constar de um comando coletivo, a execução sequencial dos comandos que o compõem é interrompida ao encontrar PAUSE; e se, em seguida a seu nome, foi digitada e gravada alguma mensagem ou instrução destinada ao operador, ela será exibida no vídeo, seguida da seguinte mensagem emitida pelo sistema operacional:

Aperte uma tecla

ou

Strike a key when ready

Se for pressionada qualquer tecla, a execução dos comandos prosseguirá normalmente.

Se forem pressionadas simultaneamente as teclas CTRL e C, surgirá no vídeo outra mensagem:

Encerra auto comando (S/N)?

ou

Terminate batch file (Y/N)?

Se a resposta for positiva, o processo será abortado. Se for negativa, a execução prosseguirá normalmente.

O comando REM tem por finalidade permitir que instruções, comentários ou lembranças apareçam no vídeo durante o processamento ou execução de um COMANDO COLETIVO, também conhecido como arquivo "batch" ou processamento em lote.

A exibição dos comentários não interrompe a execução sequencial dos comandos, como acontece com PAUSE.

Para serem válidos, os comentários devem ser digitados logo após a palavra REM.

Como separadores dos comentários são válidos apenas espaços, vírgulas e a função TAB.

~ Este comando é usado para alterar ou mudar o nomes de programas ou arquivos existentes em disquete.

Sua sintaxe é:

```
REN [d:]nomepro_antigo nomepro_novo
```

d: deverá ser indicado (DISK DRIVE A: ou B:) se o sistema estiver operando com duas unidades e o disquete que contém o programa cujo nome deve ser alterado estiver na unidade não assumida.

Os caracteres especiais * e ? podem ser usados com este comando. Se usados no nomepro antigo, todos os programas cujos nomes se ajustem a ele terão seus nomes mudados. Se usados no nomepro novo, os caracteres correspondentes às suas posições não serão mudados.

No exemplo que segue todos os programas com a extensão .BAS permanecem com os mesmos nomes, porém sua extensão será mudada para .TXT:

```
REN *.BAS *.TXT
```

No exemplo de comando apresentado a seguir, o nome do programa ou arquivo FICHAPES é alterado no disquete para DATAPES:

```
RENAME FICHAPES DATA???
```

Se nomepro novo coincidir com nome de programa já existente no disquete, o processo será abortado e será exibida a seguinte mensagem no vídeo:

Erro na troca de nomes

ou Rename error

O comando TIME é utilizado para ser exibido e/ou alterado o horário do relógio interno do computador.

Sua sintaxe é:

TIME [horas,minutos,segundos]

Quando um programa ou arquivo é gravado, o horário correspondente corrente no computador também é gravado juntamente com o nome, data e extensão do programa no diretório do disquete.

Todos esses dados são exibidos no vídeo ao ser dado o comando DIR sem qualquer opção.

Com os parâmetros opcionais especificados, o comando TIME altera o horário do relógio do computador em conformidade.

Sem os argumentos opcionais, o comando TIME origina a exibição no vídeo do horário corrente no relógio do computador e induz o usuário a alterá-lo, se desejar, como se deduz da mensagem reproduzida a seguir:

Horário corrente é hh:mm:ss:cc

Digite novo horário:

ou Current time is hh:mm:ss:cc

Enter new time

Se for apertada a tecla RETURN, será mantido o horário corrente.

Para ser mudado, o novo horário deverá ser digitado com separação de vírgulas, sendo válidos números de acordo com a seguinte tabela:

hh: 00-24

mm: 00-59

ss: 00-59

não sendo necessário digitar centésimos de segundos, enquanto que a indicação dos segundos é opcional.

Se forem digitados números em desacordo com os parâmetros indicados, o sistema operacional emitirá mensagem informando sua invalidez.

Observação: as primeiras versões de computadores padrão MSX nacionais não dispõem de relógio interno.

Sob este comando é exibido na tela de vídeo o conteúdo de um arquivo de dados ou um programa em BASIC em formato ASCII, gravados em disquete.

Sua sintaxe é:

TYPE [d:]nomepro[CONTROL+P]

d: deverá ser indicado se o disquete que contiver o arquivo a ser processado pelo comando estiver no DISK DRIVE não assumido e houver duas unidades operando no sistema.

A sequência de exibição dos dados no vídeo pode ser sustada pressionando-se simultaneamente as teclas

CTRL e S (=CONTROL e S)

Pressionando-se em seguida qualquer tecla, a exibição prossegue.

Se, antes de digitar o nomepro, ou depois de digitado mas antes de teclar RETURN, forem pressionadas simultaneamente as teclas

CONTROL e P

o conteúdo do arquivo ou do programa será passado também à impressora, sendo que os caracteres enviados para impressão serão apenas os da tabela ASCII, conforme explicado no comando COPY deste mesmo sistema.

Enquanto estiver ativada a função CONTROL+P, serão ecoados na impressora todos os comandos TYPEs dados, sendo exibidos simultaneamente no vídeo e impressos em papel os dados do programa processado.

Para anular a função ativada por CONTROL+P, devem ser pressionadas simultaneamente as teclas CONTROL e N. (Consulte a respeito "FUNÇÕES DE CONTROLE POR CARACTERE NO MSX DOS", item 10.6 deste capítulo.)

TYPE não é válido para programas gravados em formato binário ou em linguagem de máquina.

O comando VERIFY estabelece ou liga o modo "verificação" durante a gravação de dados no disquete, através do comando COPY.

Sua sintaxe é:..

VERIFY ON ou VERIFY OFF

Com o modo verificação ativado (em ON), quando é feita uma gravação no disquete, o sistema operacional verifica (ou confere) se os dados gravados coincidem com os da memória do computador. Se for constatada alguma diferença, o processo será interrompido e será exibida no vídeo a seguinte mensagem:

Erro de disco em d:

ou Bad FAT, drive d:

Devemos lembrar que, no processo de cópia de programas de um disquete para outro, os dados a serem copiados são armazenados na memória do computador, antes de serem transferidos para o outro disquete.

Com o modo VERIFY ativado, as gravações de programas ou arquivos são mais lentas do que no modo normal, porém mais confiáveis.

10.3 - AUTOEXEC.BAT - COMANDO AUTO-EXECUTANTE

É um arquivo de execução espontânea ou automática. É constituído por um ou mais comandos normais do próprio sistema MSX DOS.

Quando incluído no AUTOEXEC.BAT, qualquer comando do MSX DOS passa a ter execução automática.

Sempre que o MSX DOS é ativado, seu sistema operacional verifica espontaneamente se há no disquete do DISK DRIVE assumido um arquivo de nome AUTOEXEC.BAT. Encontrando-o, executa-o pronta e automaticamente.

Quando isso acontece, não aparece no vídeo a etiqueta de inicialização do sistema e a costumeira mensagem induzindo o operador a entrar com a data atual. Assim, se houver conveniência em atualizar sempre a data do disquete, será necessário incluir no AUTOEXEC.BAT o comando DATE como primeiro comando, ou, ao menos, em posição antecedente a qualquer comando que implique alteração no diretório do disquete.

As principais aplicações do AUTOEXEC.BAT são:

- executar automaticamente um comando do próprio sistema, ou uma série deles, quando o sistema é ativado.
- transferir automaticamente o controle do computador para o BASIC, com pronta execução de um programa do disquete.

10.3.1 - INCLUSÃO DO AUTOEXEC.BAT NO DISQUETE

O procedimento para inserção do AUTOEXEC.BAT num disquete é indicado a seguir:

- ativar o MSX DOS no modo convencional.
- digitar: COPY CON AUTOEXEC.BAT.
- pressionar a tecla RETURN.
- digitar os comandos desejados, um por vez, pressionando sempre em seguida a tecla RETURN.

- não havendo mais comandos a digitar, pressionar simultaneamente as teclas CONTROL e Z.
- pressionar mais uma vez a tecla RETURN.

Será ouvido a seguir o DISK DRIVE em movimento e aparecerá no vídeo, ato contínuo, a mensagem:

1 Arquivo(s) copiado(s)

ou 1 File(s) copied

e o AUTOEXEC.BAT estará gravado no disquete.

Note-se no segundo procedimento que a palavra CON, digitada após COPY, indica ao sistema operacional que será válido para gravação no disquete - e portanto integrando o AUTOEXEC.BAT - o que for digitado em seguida, até que sejam pressionadas simultaneamente as teclas CTRL e Z, que finalizam o processo de entrada de dados por esse processo.

Se, por exemplo, pretender-se colocar no AUTOEXEC.BAT um comando para execução automática de um programa em BASIC todas as vezes que o MSX DOS for ativado, o procedimento é o mesmo descrito acima, com apenas uma diferença: deve ser digitado somente o comando BASIC e, em seguida a ele, separado apenas por um espaço, o nome do programa desejado, como exemplificado a seguir:

- digitar: COPY CON AUTOEXEC.BAT.
- pressionar RETURN.
- digitar: BASIC NOMEPRO.
- pressionar RETURN.
- pressionar simultaneamente as teclas CONTROL e Z.
- pressionar novamente RETURN.

Assim, toda vez que o MSX DOS for ativado com o disquete que contém o AUTOEXEC.BAT gravado na disposição acima exemplificada, o sistema operacional transferirá automaticamente o controle para o BASIC e o programa NOMEPRO do disquete será executado prontamente.

NOMEPRO é o nome do programa em BASIC e pode ser qualquer palavra que não seja uma das especiais reservadas ao sistema. Após o nomepro, poderá ser usada uma extensão, na forma convencional.

Ao terminar a execução do programa NOMEPRO, o controle do computador estará com o BASIC e, por extensão, com o DISC BASIC, podendo o controle retornar ao MSX DOS através do comando CALL SYSTEM.

10.4 - COMANDO COLETIVO

OU PROCESSAMENTO EM LOTE

Além do AUTOEXEC.BAT e com a finalidade de dinamizar as operações com disquetes, o MSX DOS dispõe de um interessante recurso, que permite agrupar comandos para execução sequencial automática. Trata-se do

COMANDO COLETIVO.

Também referido como

PROCESSAMENTO EM LOTE,

o COMANDO COLETIVO é operado de maneira muito simples: primeiramente, são determinados os comandos a serem usados e definida a sequência em que deverão ser executados. Finalmente, os mesmos são digitados e gravados no disquete como se constituíssem um programa, sem numeração das linhas, e, como tal, recebendo um nome.

Para executá-lo, o procedimento é o mesmo adotado para todos os comandos do MSX DOS: estando o cursor do sistema em estado de prontidão (A>), o nome é digitado - no caso do COMANDO COLETIVO o nome sob o qual os comandos agrupados foram gravados no disquete - e pressionada uma vez a tecla RETURN.

Duas particularidades, todavia, devem ser observadas:

- o nome adotado para o comando coletivo deve ser gravado com a

extensão .BAT, que indica ao sistema operacional tratar-se de COMANDO COLETIVO, para fins de execução automática dos comandos nele contidos.

- os comandos desejados devem ser digitados e gravados no disquete através do próprio MSX DOS, usando-se o comando COPY CON e a função de controle por caractere CONTROL+Z, nos moldes da criação do comando auto-executante AUTOEXEC.BAT, já descrito, e como será mais uma vez exemplificado a seguir.

Como primeira providência, escolhemos o nome sob o qual o COMANDO COLETIVO será identificado e gravado no disquete - digamos ORDEXEC.BAT - e digitamos:

```
COPY CON ORDEXEC.BAT
```

(COPY CON indica ao sistema operacional que o que for digitado em seguida deverá ser copiado (gravado) no disquete sob o nome indicado.)

Pressionamos a tecla RETURN.

Em seguida, digitamos os nomes dos comandos do MSX DOS (devem ser apenas os dele) que desejamos sejam executados em sequência autônoma. Por exemplo:

```
TYPE MALADIR.ARQ
COPY MALADIR.ARQ PRN
COPY A:FIPESDIS.ARQ B:
DEL A:FIPESDIS.ARQ
TYPE B:FIPESDIS.ARQ
BASIC FIPESDIS.BAS
```

Após cada comando digitado, deve ser pressionada a tecla RETURN.

Quando não houver mais linhas de comandos a digitar, deve ser finalizado o processo de entrada pressionando-se concomitantemente as teclas CONTROL e Z.

Em seguida, pressionamos novamente a tecla RETURN. Será ouvido o acionamento automático do DISK DRIVE assumido e o COMANDO COLETIVO de nome ORDEXEC.BAT estará gravado no disquete e pronto a ser usado.

Para pôr em execução o comando coletivo assim criado, bastará digitar seu nome - sem a extensão .BAT - quando o cursor do sistema (A>) estiver aguardando comando. Por exemplo:

A> ORDEXEC

Tecler RETURN e todos os comandos sob o nome ORDEXEC.BAT serão automaticamente executados! Obviamente, no exemplo acima, os disquetes que contêm os programas designados no comando coletivo deverão estar devidamente alojados, isto é, o disquete que contém o ORDEXEC.BAT e os demais programas envolvidos no processamento deverá estar no DISK DRIVE A e o disquete que irá receber a cópia do arquivo FIPESDIS.ARQ deverá estar no DISK DRIVE B.

No caso de o computador estar operando com um DISK DRIVE apenas, o sistema operacional indicará através do vídeo quando deverá ser efetuada a troca de disquetes, isto é, quando deverá ser alojado no DISK DRIVE A o disquete correspondente ao B. Outras trocas eventuais também serão indicadas pelo sistema operacional.

A impressora deverá estar pronta para imprimir os dados do arquivo MALADIR.ARQ. Caso se deseje aprontar a impressora apenas no momento de execução do comando correspondente, deverá ser ele precedido por um comando PAUSE devidamente complementado pela lembrança adequada.

O COMANDO COLETIVO exemplificado acima poderá também ser gravado com o nome AUTOEXEC.BAT, caso em que entrará automaticamente em execução sempre que o MSX DOS for ativado com o disquete em que estiver ele presente.

A execução sequencial do comando coletivo poderá ser interrompida pressionando-se simultaneamente as teclas CTRL e C, caso em que será emitida pelo sistema operacional, através do vídeo, a mensagem:

Encerra auto comando (S/N)?

ou

Terminate batch job (Y/N)?

Se for pressionada a tecla S (ou Y), os comandos ainda não executados serão ignorados - o processo será abortado - e o cursor do sistema reaparecerá no vídeo em prontidão para comando.

Se for pressionada a tecla N, a execução prosseguirá a partir do comando seguinte ao que foi interrompido, sem ser executado o remanescente deste.

Durante a execução de um comando coletivo, se for desalojado do DISK DRIVE o disquete que o contém, o sistema induz o operador a alojá-lo novamente, para que o próximo comando possa ser lido, através da seguinte mensagem:

Insira disco c/arquivo batch

Aperte uma tecla

ou Insert disk with batch file

Strike any key when ready

O último comando de um comando coletivo pode ser outro da mesma modalidade, permitindo assim que nova série seja executada sem a interferência do usuário ou operador.

Na secção de comandos do MSX DOS são descritos dois de aplicação exclusiva em comandos coletivos: PAUSE e REM. Recomendamos revê-los.

10.4.1 - PARÂMETROS SUBSTITUÍVEIS OU

"MANEQUINS" NOS COMANDOS COLETIVOS

Os comandos coletivos podem ser criados com parâmetros substituíveis ou "manequins", isto é, com opções de comandos ou de programas, os quais são definidos pelo operador no momento de ser dado o comando para execução do conjunto.

Tal recurso permite que um mesmo comando coletivo seja utilizado para processamento diversificado.

Os parâmetros substituíveis são dez e identificados para o sistema operacional pelo sinal de porcentagem (%), seguido de um número que pode variar de 0 a 9: %0, %1, %2 etc.

Como exemplo, vamos criar um comando coletivo que possa ser utilizado para processamento de diversos outros arquivos individuais ou grupos de dados.

Seja um sistema de "mala direta", em que os destinatários estejam agrupados por Estado, num mesmo disquete. Teremos então:

MALDIRSP.ARQ

MALDIRGO.ARQ

MALDIRPR.ARQ

...

O processamento do sistema deve ser de tal modo que, em função de determinados fatores, apenas um Estado seja selecionado por vez para expedição da "mala direta".

Depois de determinado o Estado, o arquivo de dados correspondente deve ser exibido no vídeo, para verificação da necessidade eventual de alterações ou cancelamentos.

Em seguida, esse arquivo de dados deve ser copiado em outro disquete e eliminado do original.

Finalmente, as etiquetas correspondentes devem ser emitidas e impressas, devendo a impressora ser ligada no momento de execução do comando específico.

Podemos deduzir facilmente que, estando programados doze Estados, por exemplo, para envio de "mala direta", necessitaríamos de doze programas para o processamento descrito, se não contássemos com o recurso dos parâmetros substituíveis ou "manequins".

Utilizando os "manequins", podemos criar um único comando coletivo para a mesma finalidade, como o exemplificado a seguir:

Nome do comando:

MALDIRBR.BAT

Comandos a incluir:

```
TYPE %1.ARQ
COPY B:%1.ARQ
DEL A:%1.ARQ
PAUSE APRONTE A IMPRESSORA ETC.
COPY B:%1.ARQ PRN
```

Para fazê-lo "rodar" e executar todos os seus comandos apenas em relação ao arquivo de um determinado Estado, basta simplesmente digitar o nome dado ao comando coletivo e logo em seguida a ele, separado apenas por um espaço, o nome do arquivo correspondente ao Estado, como no exemplo que segue:

```
MALDIRBR MALDIRSP.ARQ
```

(Lembramos que, em se tratando de programa ou arquivo que tenha a extensão .BAT, esta não deve ser digitada quando é dado o comando para sua execução.)

Nessa condição, o parâmetro substituível %1 constante dos comandos incluídos no comando coletivo MALDIRBR.BAT será substituído automaticamente pelo nome do arquivo do Estado determinado, tendo execução sequencial normal.

No exemplo citado, há apenas um "manequim", %1, que é substituído por um arquivo de dados.

Se forem usados diversos "manequins" ou parâmetros substituíveis, os arquivos que devem entrar em seu lugar devem ser digitados na mesma linha de comando, um após outro, separados apenas por um espaço.

O parâmetro %0 só serve para ser substituído pelo próprio comando coletivo. Por exemplo, se ao comando coletivo MALDIRBR.BAT for acrescentado como último comando

```
TYPE %0
```

todos os comandos nele agrupados serão listados no vídeo depois de serem executados sequencialmente.

No exemplo acima usamos "manequins" para serem substituídos por nomes de arquivos. Usaremos a seguir "manequins" para serem substituídos por comandos do próprio sistema operacional.

Suponhamos que certos arquivos de uma empresa devam ter seus registros examinados regularmente e que, ocasionalmente, tais registros tenham que ser listados por impressora para exame mais minucioso e outras providências.

Em vez de ter dois comandos coletivos com comandos diferentes para os mesmos arquivos individuais, um único programa pode servir para as duas finalidades e outras, se forem utilizados "manequins" ou parâmetros substituíveis. Por exemplo, reunindo os seguintes programas:

AGENDA.ARQ

CTASPGAR.ARQ

CTASREC.ARQ

num comando coletivo de nome

COMPROM.BAT

e usando apenas dois parâmetros substituíveis, digamos %1 e %2, o coletivo seria criado na seguinte forma:

%1 AGENDA.ARQ %2

%1 CTASPGAR.ARQ %2

%1 CTASREC.ARQ %2

Assim, se se desejar apenas examinar seus registros no vídeo, deve ser dado o comando

COMPROM TYPE NUL

Se se pretender tê-los copiados pela impressora, o comando será

COMPROM COPY PRN

Como foi visto no item 8.1, "SINTAXE DOS COMANDOS", a palavra NUL é argumento NULO para qualquer dispositivo ou opção de comando.

10.5 - CHAVES ESPECIAIS DE EDIÇÃO

A fim de permitir maior agilização na manipulação ou controle de seus comandos, o MSX DOS é provido de recursos denominados CHAVES ESPECIAIS DE EDIÇÃO.

São assim denominados porque permitem que:

- uma linha de comando possa ser instantaneamente repetida, pressionando-se apenas duas teclas na maioria das vezes, sendo uma delas a tecla RETURN;
- uma linha de comando, digitada com erro e invalidada, possa ser editada e corrigida sem que seja necessário redigitá-la completamente;
- uma linha de comando similar a uma precedente seja editada para execução com o mínimo necessário de alterações.

As chaves especiais de edição merecem particular ênfase pela agilização que conferem aos comandos e pela comodidade que oferecem ao operador. Sua eficiência é devida ao fato de atuarem no setor do sistema operacional em que é controlada a maioria dos comandos.

Sabendo como funciona o sistema, poderemos fazer melhor uso dos recursos que oferece. Vamos tentar entendê-lo:

Ao ser ativado, o MSX DOS cria uma área especial de memória, denominada "template".

Nessa área é sempre armazenada a cópia da última linha de comando dado ao sistema operacional, isto é, uma linha de comando que já recebeu a ordem de execução ao ser pressionada a tecla RETURN.

Assim, através do uso de teclas designadas no sistema operacional para funcionarem como

CHAVES ESPECIAIS DE EDIÇÃO

a linha armazenada no "template" é editada e tratada, ou seja: é alterada, corrigida ou simplesmente repetida, recebendo a seguir nova or-

dem de execução pressionando-se a tecla RETURN.

Para facilidade de acesso e consultas, todas as chaves especiais de edição se encontram reunidas na

"TABELA DAS CHAVES ESPECIAIS DE EDIÇÃO DO MSX DOS",

no apêndice, no final do livro.

A seguir, são apresentados exemplos de uso das chaves especiais de edição.

10.5.1 - EXEMPLOS DE USO DAS

CHAVES ESPECIAIS DE EDIÇÃO

Dando-se o comando

TYPE MALDIR2.ARQ

os registros do arquivo gravado num disquete com esse nome serão exibidos no vídeo e a linha de comando TYPE MALDIR2.ARQ será armazenada no "template".

Para repetir o comando, basta pressionar a tecla de cursor para baixo (chave COPYALL) e em seguida a tecla RETURN.

Se se desejar dar o mesmo comando para o arquivo MALDIR3.ARQ, deverão ser pressionadas as teclas SELECT e 2 (chave COPYUP) para surgir na linha de comando apenas a palavra MALDIR, ficando o cursor posicionado logo após o caractere R. Digitando-se em seguida o caractere 3 e pressionando-se a tecla de cursor para baixo (v), a linha será completada com .ARQ, restando pressionar apenas RETURN para que seja executado o comando. (O número 2 teclado juntamente com SELECT indicou ao sistema que a linha de comando copiada no "template" deveria ser editada até aquele número, exclusive.)

Vamos agora corrigir a linha de comando digitada como

quando deveria ser

TYPE FICHAPES.ASC

Para melhor compreensão, indicaremos apenas as teclas que devem ser pressionadas e, entre parênteses, o nome da chave de edição a ser usada:

T (COPY1) P (COPYALL)

(Se a tecla RETURN não tiver sido pressionada, a linha de comando ainda não foi armazenada no "template". Nesse caso, para que a linha seja armazenada no "template" sem ser efetivado o comando e possa ser feita a experiência de correção, deve ser usada a chave NEWLINE.)

A primeira vista, as chaves especiais de edição não causam a real impressão de sua validade. Mas, sobretudo para quem usa intensivamente o sistema operacional para disco, sua utilização é inestimável e poderá ser facilmente constatada após experiências práticas que demonstrem toda sua potencialidade.

10.6 - FUNÇÕES DE CONTROLE POR

CARACTERE NO MSX DOS

Função de controle por caractere é uma função do MSX que afeta a linha de comando e/ou o comando propriamente dito, complementando-os em sua digitação ou execução.

No MSX DOS as funções de controle por caractere são ativadas pela tecla CONTROL (ou CTRL) pressionada simultaneamente com uma das teclas seguintes:

C, H, J, M, N, P e S

Por exemplo, durante a execução de um comando, se forem pressio-

nadas ao mesmo tempo as teclas

CONTROL e C

a execução será interrompida e cancelada.

Se forem pressionadas juntamente as teclas

CONTROL e P

será ativado o sistema "eco" do sistema operacional e o que for exibido no vídeo, por força de execução de comandos, será também impresso através da impressora, se estiver devidamente conectada ao computador.

Para facilidade de acesso e consultas, todas as funções desta categoria são descritas na "TABELA DAS FUNÇÕES DE CONTROLE POR CARACTERE DO MSX DOS, no Apêndice, no final do livro.

10.7 - OCORRÊNCIAS DE ERROS EM OPERAÇÕES DO MSX DOS E MENSAGENS DO SISTEMA

Durante a execução de um comando, quando ocorre um tipo de erro inerente ao próprio disquete - denominado "erro de disco" -, o MSX DOS tenta três vezes, automaticamente, completar a operação. Se não conseguir, interrompe o processo e coloca no vídeo uma mensagem para o operador, reportando a ocorrência, e que pode ser uma das seguintes:

Erro de disco p/leitura (/escrita) em d:

ou Disc error reading (/writing) drive d:

Disco protegido p/escrita em d:

ou Write protect error writing drive d:

```

ou          Not ready error reading
          (/writing) drive d:

```

Cancela, Repete ou Ignora?

ou Abort, Retry, Ignore?

Usualmente, o operador tenta repetir a operação e, não conseguindo sucesso, cancela o comando.

Erro de disco em d:

ou Bad FAT drive d:

indicando falta de formatação do disquete ou falha na formatação ou no diretório. Nestes casos, geralmente o disquete só pode ser reutilizado após nova formatação, se não houver falha material no mesmo.

Diversos outros erros originam a emissão de mensagens do sistema operacional para o operador, através do vídeo. Geralmente são erros de comando ou falhas do operador, como digitar errado um nome de comando ou de programa ou arquivo, indicar um parâmetro inválido, esquecer de alojar no DISK DRIVE o disquete certo etc.

Por outro lado, algumas mensagens são emitidas apenas a título de apoio às operações do sistema ou para orientação do usuário.

No final do livro, no Apêndice, é encontrada a "TABELA DE MENSAGENS DO MSX DOS", onde são enumeradas todas as mensagens emitidas pelo sistema operacional, com seus respectivos significados.

1.1 - COMANDOS E RECURSOS

DO HB-MCP

Existem dois tipos de comandos no sistema operacional para disco HB-MCP, exclusivo do microcomputador HOTBIT: comandos internos e comandos externos.

Os comandos internos, também denominados comandos residentes no HB-MCP, são os que fazem parte integrante do programa que constitui o sistema propriamente dito. Estes comandos não são visualizáveis no vídeo sob o comando DIR, por exemplo, e se tornam automaticamente disponíveis ao usuário quando o sistema é ativado.

Os comandos externos, também denominados comandos temporários ou utilitários, não estão incluídos no programa de sistema propriamente dito. Residem separadamente no disquete e são carregados para a memória do computador e executados quando seus nomes são digitados na forma de comando. Depois de executados, são eliminados da memória do computador, só retornando a ela sob nova invocação. Em verdade, são programas utilitários que funcionam como comandos do sistema operacional, completando-o com maior versatilidade.

Diferentemente do DISC BASIC E DO MSX DOS, o HB-MCP permite que sejam dados diversos comandos por vez, desde que digitados em sequência numa mesma linha - até 254 caracteres, incluindo espaços em branco - e separados por vírgulas.

11.1 - TABELA DOS COMANDOS DO HB-MCP

COMANDOS INTERNOS OU RESIDENTES

DIR	REN	TYPE
ERA	SAVE	USER

COMANDOS EXTERNOS OU TEMPORÁRIOS

AUTOLD	DROUT	STAT
BACKUP	DSKCNV	SUBMIT
BATCH	DUMP	SYSGEN
COPDOS	FORMAT	TERM
COPY	FUNÇÕES	XSUB
DRINP	PIP	

11.2 - DESCRIÇÃO DOS COMANDOS DO HB-MCP

A fim de facilitar o acesso aos mesmos, para fins de consulta, os comandos do HB-MCP serão apresentados em ordem alfabética, sem separação de tipo, cada um sempre começando em página nova, a partir da seguinte, e com seus nomes em destaque.

Após a descrição dos comandos serão apresentados outros recursos do sistema.

É um utilitário auto-executante, isto é, sua execução é automática quando o sistema HB-MCP é inicializado com o disquete que o contém no formato ativo.

Para estar no formato ativo, o AUTOLD deve ser gravado no disquete com comandos do HB-MCP, os quais, por constarem dele, serão também executados automaticamente e na sequência em que foram inseridos.

Sua sintaxe é:

```
AUTOLD [comando][,comando][,comando]
```

Podem constar do AUTOLD tantos comandos quantos couberem em uma linha de comando, até o limite de 254 caracteres.

Para ser auto-executante, o AUTOLD deve ser gravado no disquete através do procedimento indicado a seguir:

- estando o cursor do sistema em prontidão de comando, digita-se a palavra AUTOLD e os comandos desejados, na disposição indicada em sua sintaxe. Exemplo:

```
AUTOLD DIR,STAT TEXTART1.BAS,TYPE CHAPT10-5.ARQ,COPDOS  
ATLASPRO.BAS,ERA MALDIRSP.ARQ,STAT *.*
```

- durante a digitação de comandos, as mudanças necessárias de linha do cursor devem ser efetuadas através da função CONTROL+E, a fim de que o sistema operacional interprete corretamente todo o elenco de comandos.

Para ser desativado, os comandos inseridos no AUTOLD devem ser apagados do disquete, o que é feito digitando-se apenas o nome AUTOLD após o cursor em prontidão para comando e teclando-se RETURN.

Em ambos os casos, o sistema operacional solicitará confirmação do operador para gravação no disquete, através da seguinte mensagem:

AUTOLD

Auto-execução de comando

V1.0

Comando:

Pode gravar no disco (S/N) ?

(Quando são digitados comandos para entrada, eles serão exibidos no quadro, antes da solicitação de confirmação ou não para gravação.)

A eliminação dos comandos inseridos do AUTOLD é considerada pelo sistema operacional também como um processo de gravação - ou anulação da gravação -, o que explica a ambiguidade da mensagem.

A seguir, um exemplo de comando AUTOLD, para teste imediato:

```
AUTOLD DIR *.COM, STAT *.COM
```

Gravado dessa forma num disquete, toda vez que o HB-MCP for ativado através do disquete que o contiver, serão listados no vídeo todos os programas e arquivos cujos nomes tenham a extensão .COM, e, em seguida, serão apresentados os dados dos mesmos.

IMPORTANTE:

O AUTOLD não verifica o tipo de disquete alojado no DISK DRIVE corrente. Se o mesmo não for do padrão HB-MCP, sofrerá danos e nenhuma mensagem da ocorrência será emitida pelo sistema.

Sendo um dos comandos externos do HB-MCP, o BACKUP é o utilitário que copia disquetes "fisicamente", isto é, ele efetua a cópia do disquete trilha por trilha, sequencialmente, a partir do início de cada uma, o que torna o processo um tanto lento quando o sistema opera com apenas um DISK DRIVE.

Sua sintaxe é:

BACKUP [d:destino=d:origem]

Quando não especificados d:destino e d:origem, o sistema executa a operação toda no DISK DRIVE A, independentemente de haver dois operando e de o corrente ser o B, por exemplo.

Em tal caso, e também no caso de haver um único DISK DRIVE, quando então o comando a ser dado é simplesmente:

BACKUP

o sistema operacional indica ao operador, através do vídeo, as trocas que devem ser efetuadas entre os disquetes origem e destino.

Como o processo é de cópia "física", podem ser efetuadas cópias de disquetes de quaisquer formatos, desde que o padrão (faces/trilhas/densidade) seja o mesmo.

O processo deve sempre ser iniciado com o disquete que contém o sistema HB-MCP. Depois de iniciado, o operador é orientado pelo sistema operacional, através do vídeo, sobre as trocas de disquetes que devem ser feitas.

A operação pode ser interrompida, pressionando-se simultaneamente as teclas

CONTROL e C

Incluído também no conjunto de comandos externos ou temporários, BATCH é um utilitário do HB-MCP que permite que uma série de comandos seja executada automaticamente, logo após sua digitação.

Sua sintaxe é:

BATCH

=comando

=comando

...

Depois de dado o comando BATCH, os outros comandos são solicitados pelo sistema operacional com a aposição, na linha seguinte, do sinal de igualdade, após o qual deve ser digitado o nome do comando desejado, repetindo-se sucessivamente o processo.

Para iniciar a execução da série de comandos, basta teclar RETURN quando é aposto um novo sinal de igualdade.

Para interromper a entrada de comandos e também cancelar sua execução, devem ser pressionadas simultaneamente as teclas

CONTROL e C.

Ao contrário do comando SUBMIT, também com função de execução coletiva de comandos, a série de comandos criada com BATCH não é gravada no disquete como um arquivo, sendo descartada depois de sua execução.

Exemplo de comando BATCH:

BATCH

=STAT *.*

=TYPE CHAPT10.ARQ

=

/RETURN

O comando utilitário COPDOS copia programas e arquivos de disquetes do sistema HB-MCP em disquetes do sistema MSX DOS, assim como copia programas e arquivos de disquetes do sistema MSX DOS em disquetes do sistema HB-MCP, não obstante ser diferente a formatação de ambos.

Sua sintaxe é:

COPDOS nomepro

A operação é feita com apenas um DISK DRIVE e é iniciada sempre com o disquete que contém o HB-MCP, sendo a permutação necessária dos disquetes orientada pelo sistema através do vídeo.

São válidos em nomepro os caracteres especiais \$ e ?.

Ao ser iniciado o processo, é exibida a mensagem:

COPDOS

Intercâmbio HB-MCP/HB-DOS

V1.0H

(1) HB-MCP para HB-DOS

(2) HB-DOS para HB-MCP

Escolha opção:

(HB-DOS é nome registrado da EPCOM EQUIPAMENTOS ELETRÔNICOS LTDA. para seu sistema equivalente ao MSX DOS.)

Indicada a opção ao sistema, surge no vídeo a mensagem:

Coloque disco origem

e tecla <RETURN>

Se nomepro especificado no comando foi TEXTART?.BAS, por exemplo,

e houver no disquete origem dois programas que se enquadrem nessa especificação, o restante do processo se desenvolverá de acordo com as mensagens reproduzidas a seguir e que são emitidas pelo sistema operacional:

TEXTART1.BAS - leitura completa

TEXTART2.BAS - leitura completa

Coloque disco destino

e tecle <RETURN>

TEXTART1.BAS - gravação completa

TEXTART2.BAS - gravação completa

Fim de cópia

Coloque disquete com sistema

e tecle <RETURN>

Se ocorrer alguma anomalia durante o desenvolvimento do processo, mensagens correspondentes serão emitidas pelo sistema operacional.

A finalidade do comando utilitário COPY é executar cópias de programas e arquivos de um disquete para outro, utilizando apenas um DISK DRIVE. Sua sintaxe é:

COPY [nomepro]

Se não for especificado nomepro, todos os programas e arquivos do disquete origem serão copiados no disquete destino. Em tal caso, será solicitada confirmação pelo sistema operacional e as cópias serão feitas no disquete destino enquanto houver espaço disponível nele.

Se for encontrado no disquete destino um programa ou arquivo com o mesmo nome do que vai ser copiado nele, será exibida no vídeo a mensagem reproduzida a seguir:

Arquivo existe. Elimina (S/N) ?

Digitando-se "S", será eliminado o já existente e feita a cópia do novo. "N" preservará o já existente, não copiando o novo.

As trocas necessárias dos disquetes no DISK DRIVE são solicitadas pelo sistema operacional, através do vídeo.

Os caracteres especiais * e ? podem ser usados em nomepro, permitindo a execução seletiva de grupos de programas ou arquivos.

Se estiver configurado o DISK DRIVE E (ver o comando DSKCNV sobre o assunto) e for dado o comando COPY, o sistema operacional solicitará definição de opção de acordo com o "menu" reproduzido a seguir:

COPY

Copia Arquivos

V1.0

Drive Origem

Drive Destino

(1)	A:	A:
(2)	A:	E:
(3)	E:	A:
(4)	E:	E:

Escolha opção:

Nesta situação, é possível efetuar cópias de programas e arquivos entre disquetes com formatações diferentes, desde que compatíveis com o CP/M.

Todo o desenrolar do processo COPY é monitorado no vídeo com mensagens e solicitações de trocas dos disquetes origem e destino.

Se ocorrer algum erro durante o processo de execução de cópia, o sistema operacional emitirá mensagens de orientação. (Ver a "TABELA DE MENSAGENS DO HB-MCP", no Apêndice, no final do livro.

Exemplos de comandos:

COPY

COPY *.ARQ

COPY TEXT??.BAS

A finalidade do comando DIR é exibir no vídeo o diretório do disquete. Sua sintaxe é:

DIR [d:][nomepro]

Havendo dois DISK DRIVES em operação, d: deverá ser especificado se o comando for dirigido a disquete que não se encontrar na unidade assumida ou corrente.

Se nomepro não for especificado no comando, todos os nomes de arquivos e programas existentes no disquete serão listados no vídeo.

Os caracteres especiais * e ? podem ser usados em nomepro, nas mesmas condições em que são usados no DISC BASIC.

Diferentemente do equivalente do MSX DOS, o comando DIR do HB-MCP não indica a extensão em "bytes" do(s) programa(s) e arquivo(s) listado(s), nem o espaço livre disponível no disquete. Indica apenas o DISK DRIVE corrente, na frente de cada linha listada.

O comando DIR do HB-MCP não é válido para o MSX DOS e vice-versa, em vista das formatações diferentes dos disquetes dos dois sistemas.

É apresentada a seguir uma listagem típica feita pelo comando DIR sem parâmetros:

A: AUTOLD	COM : BATCH	COM
A: BACKUP	COM : COPDOS	COM
A: COPY	COM : DRINP	COM
A: DROUT	COM : DSKCNV	COM
A: FORMAT	COM : FUNÇÕES	COM
A: PIP	COM : STAT	COM
A: SUBMIT	COM : SYSGEN	COM
A: TERM	COM : XSUB	COM
A: DUMP	COM : TEXTART1	BAS
A: CHAPT9-5	ARQ : FIPESDIS	BAS

DRINP altera ou compatibiliza a configuração dos caracteres recebidos pelo computador através de interface serial ou outro periférico. A interface serial deve ter padrão HOTBIT.

DROUT altera ou compatibiliza a configuração dos caracteres gerados no computador, para saída para interface serial, impressora ou outro periférico.

Computador, impressora e interface serial geram e reproduzem caracteres - letras, números, símbolos - de acordo com padrões estabelecidos em tabelas existentes para essa finalidade.

Quando as tabelas adotadas são as mesmas, um caractere "A" gerado no computador, por exemplo, será reproduzido assim pela impressora.

Em caso contrário, poderá ser reproduzido pela impressora outro caractere, como um "Ã", por exemplo, ou mesmo um espaço em branco.

Para contornar tais problemas, os utilitários DRINP e DROUT compatibilizam os padrões de caracteres nas operações de entrada e saída entre computador e periféricos quando o HB-MCP está ativo.

As tabelas enquadradas nas conversões efetuadas pelos utilitários DRINP e DROUT pertencem aos seguintes padrões:

- ABICOMP
- ABNT
- ASCII
- MSX

O padrão ABICOMP é o "default" ou inicial adotado no HOTBIT versão 1.1 e parcialmente na versão 1.0 do EXPERT.

O padrão ABNT é o "default" ou inicial adotado na segunda versão do HOTBIT, 1.2, e na versão 1.1 do EXPERT.

O padrão ASCII é considerado para fins de conversão de programas ou arquivos desenvolvidos em sistemas que o utilizam exclusivamente.

O padrão MSX é adotado opcionalmente, como saída, tanto no HOTBIT como no EXPERT.

Geralmente, os caracteres compreendidos entre os códigos 32 e 127 (&H20 e &H7F) são coincidentes nas três tabelas, seguindo a principal que rege a matéria, a praticamente universal TABELA ASC II. Os demais caracteres variam de acordo com o padrão do computador.

Ao ser dado o comando DRINP, é exibido no vídeo o "menu":

DRINP

Converte caracteres na entrada

V1.0

- (A) Espaços para códigos acima de 7FH
- (B) Padrão ABNT ("default" no HB-MCP)
Padrão HOT-BASIC V1.2
- (C) Padrão HOT-BASIC V1.1
- (D) Padrão MSX ou Relação direta de c
ódigos
- (Z) Não altera

Opção:

A opção deve ser indicada de acordo com o equipamento utilizado.

O "menu" de DROUT é o mesmo, sendo apenas alterado o título para DROUT e mudada a palavra "entrada" para "saída".

A configuração escolhida é perdida ao desligar-se o computador.

O comando utilitário DISKCNV é um conversor de DISK DRIVE, como sugere seu nome.

Ao ser ativado, exibe no vídeo o seguinte "menu":

DISKCNV

Converte Drive Lógico

V1.0

(A) 5.25 FS HB8000

(B) 5.25 FD HB8000

(C) 3.50 FS HB8000

(D) 3.50 FD HB8000

(E) 5.25 FS S700

(F) 5.25 FD S700

(G) 5.25 FS I7000

(H) 5.25 FD I7000

(I) 5.25 FD MZ3500

(J) 5.25 FD SVI707

(Z) Não configura

Escolha opção:

Digitada a opção escolhida, um novo DISK DRIVE passa a integrar o sistema operacional HB-MCP: o DISK DRIVE LÓGICO E, embora não haja nenhum acréscimo de unidade física operacional de disco.

Sabendo-se como um DISK DRIVE é classificado no sistema operacional, entende-se o fato:

Quando o sistema opera com um único DISK DRIVE físico, este funciona e é classificado como

DRIVE FÍSICO 0 = DRIVE

OPERACIONAL A = DRIVE LÓGICO A

Quando o sistema opera com dois DISK DRIVES físicos, estes são:

DRIVE FÍSICO 0 = DRIVE

OPERACIONAL A = DRIVE LÓGICO A

DRIVE FÍSICO 1 = DRIVE

OPERACIONAL B = DRIVE LÓGICO A

Quando convertido pelo utilitário DSKCNV em DISK DRIVE LÓGICO E, um único DISK DRIVE operando no sistema (DRIVE FÍSICO 0) não perde sua condição anterior. Ele passa a ter duas funções operacionais, A e E, e duas lógicas, A e E.

Se o sistema operar com dois DISK DRIVES físicos, o utilitário DSKCNV confere a duplicidade de funções ao DRIVE FÍSICO 1, que passa a ter duplas funções operacionais, B e E, e lógicas, A e E. Neste caso, o DRIVE FÍSICO 0 continua na sua condição normal.

Condição física e função operacional de um DISK DRIVE dispensam novos comentários.

Função lógica de um DISK DRIVE é aquela que o enquadra como funcional dentro dos padrões e parâmetros de um sistema operacional específico para discos.

Tais padrões e parâmetros são constituídos pela configuração própria do disquete (diâmetro, faces, trilhas, setores) e elementos que definem posição de diretório, informações contidas no mesmo, identificação e distribuição de dados, arquivos ou programas nos espaços físicos do disquete, e diversos outros elementos.

Assim, a condição operacional lógica de um DISK DRIVE do sistema HB-MCP é normalmente "A" porque opera dentro dos padrões e parâmetros desse sistema.

Passa a ser "E", através do utilitário DSKCNV, em condição ESPECIAL ou EXCEPCIONAL, porque, depois de escolhida uma das opções oferecidas, o mesmo DISK DRIVE é configurado para funcionar também dentro dos padrões e parâmetros do sistema operacional que lhe corresponder.

Depois de estar configurado para operar em dois sistemas, a condição funcional assumida ou corrente do DISK DRIVE 0 e único é "A", e para operar como "E" deve ser especificado no comando. Exemplo:

DIR E:

No caso de haver dois DISK DRIVES operando, a condição funcional corrente ou assumida da unidade 1 é "B". Para operar como "E", também deve ser especificado no comando.

As opções oferecidas pelo DSKCNV são todas compatíveis com o CP/M e HB-MCP e identificáveis por suas referências:

- 1ª coluna: as letras indicam as opções.
- 2ª coluna: os números indicam o diâmetro do disquete.
- 3ª coluna: as abreviações FS e FD significam FACE SIMPLES e FACE DUPLA.
- 4ª coluna: as siglas indicam os sistemas operacionais que são aceitos pelo utilitário.

As opções indicadas como HB8000 referem-se ao sistema HB-MCP que roda no HOTBIT HB-8000 e nada têm a ver com o MSX DOS ou HB-DOS. Disquetes do MSX DOS e do HB-DOS não são utilizáveis sob a configuração "E", uma vez que seus parâmetros são diferentes.

A configuração feita pelo utilitário DSKCNV é perdida quando se desliga o computador.

Através do comando utilitário DUMP, programas e arquivos do disquete são listados em códigos hexadecimais e caracteres MSX correspondentes, no vídeo ou na impressora.

Sua sintaxe é:

DUMP [d:]nomepro[opção]

A listagem é feita em blocos - ou "páginas" - que correspondem ao enchimento da tela de vídeo e permite ver como o programa ou arquivo está gravado no disquete.

OPÇÃO \$D - Exibe os códigos e caracteres no vídeo.

OPÇÃO \$I - Envia os códigos e caracteres para a impressora.

Se não for indicada uma opção no comando, será adotada automaticamente pelo sistema operacional a opção \$D.

Interrompe-se a listagem pressionando-se juntamente as teclas

CONTROL e C.

Depois de preenchida a tela de vídeo, novos blocos ou páginas do programa ou arquivo são listados pressionando-se qualquer tecla.

Algumas linhas de um bloco listado são reproduzidas em seguida:

DUMP V1.0 [Arquivo = CAPT10-9.ARQ]

```
0000:2E 2E 2E 0D 0A 52 45 4D .....REM
0008:20 20 20 20 20 20 20 20
0010:20 20 20 20 20 20 20 20
0018:20 20 4D 53 58 20 44 4F   MSX DO
0020:53 0D 0A 0D 0A 0D 0A 0D S.....
0028:0A 20 20 20 20 20 4F 20   .    0
0030:63 6F 6D 61 6E 64 6F 20 comando
0038:52 45 4D 20 74 65 6D 20 REM tem
0040:70 6F 72 20 66 69 6E 61 por fina
0048:6C 69 64 61 64 65 20 70 lidade p
```

A finalidade do comando ERA é eliminar programas e arquivos do disquete.

Sua sintaxe é:

ERA [d:]nomepro

O DISK DRIVE deverá ser indicado se houver duas unidades operando no sistema e o disquete a ser processado se encontrar na unidade não corrente ou não assumida.

Os caracteres especiais * e ? podem ser usados em nomepro.

Se, em vez de nomepro, o parâmetro for indicado como segue:

ERA *.*

todos os programas e arquivos do disquete serão eliminados depois de confirmado o comando.

Tal confirmação é solicitada através do vídeo com a seguinte mensagem do sistema operacional:

Todos (S/N) ?

O uso combinado dos dois caracteres especiais * e ? permite eliminar seletivamente grupo(s) de programas e arquivos sob um único comando. Por exemplo, o comando:

ERA TXT??.*

eliminará do disquete todos os programas ou arquivos cujas três letras primeiras de seus nomes sejam TXT, sendo quaisquer a quarta, a quinta e as extensões adotadas para os nomes.

Prepara o disquete para ser utilizado no sistema operacional para discos HB-MCP.

Sua sintaxe é:

FORMAT [d:]

A formatação de disquete no HB-MCP é feita através de um programa específico do sistema e não através da interface para discos, como é o caso do DISC BASIC e do MSX DOS.

A formatação do HB-MCP é compatível com a do sistema CP/M, sendo diferente do DISC BASIC e do MSX DOS. Suas características são:

- diretório para 64 entradas de 32 "bytes" cada, ocupando o bloco 0 do disquete, começando no primeiro setor da quarta trilha;
- cada entrada no diretório pode conter informações sobre até 32 "kbytes" de espaço ocupado no disquete, ou de duas extensões;
- bloco de dados = 16 registros = 2 "kbytes";
- cada registro = 128 "bytes";
- cada extensão = 8 blocos de dados = 16 "kbytes";
- as três primeiras trilhas do disquete são sempre reservadas aos programas que constituem o HB-MCP, excetuados os referentes aos comandos externos ou utilitários;
- cada programa ou arquivo pode ocupar tantas entradas do disquete quantas forem necessárias, dependendo de sua extensão.

Uma "entrada" no disquete contém informações específicas sobre um programa ou arquivo nele armazenado. Tais informações abrangem nome do programa ou arquivo, extensão do nome, área ocupada no disquete e sua localização, quantidade de registros utilizados, alocação de blocos, classificação do conteúdo etc.

O comando FORMAT deve ser dado com o disquete que contém o siste-

na HB-MCP alojado no DISK DRIVE corrente.

O processo todo de formatação é monitorado no vídeo e o operador é orientado quanto a informações a fornecer ao sistema operacional por intermédio do teclado.

As mensagens emitidas pelo sistema depois de dado o comando e durante o desenrolar do processo são reproduzidas a seguir:

FORMAT

Inicia disquetes

V1.0

Coloque disquete para ser

formatado no drive A:

Tecle <RETURN>

Confirma (S/N) ? S

Formatando trilha 001, 002... 079

Formatação completa

Verifica (S/N) ? S

Verificando trilha 001, 002... 079

Verificação completa

Formata outro (S/N) ? N

Coloque disquete com sistema

Tecle <RETURN>

O comando FORMAT poderá também ser dado para o DISK DRIVE lógico E, desde que sua configuração tenha sido feita pelo utilitário DSKCNV.
Exemplo:

FORMAT E:

FUNÇÕES é um comando externo do HB-MCP que lista no vídeo as funções correntes das dez teclas de função programável.

Sua sintaxe é o próprio nome:

FUNÇÕES

É um comando utilitário, que pode ser ativado na forma convencional, isto é, digitando seu nome e teclando RETURN em seguida, ou simplesmente pressionando a tecla F1, após o que o vídeo exibirá:

FUNÇÕES

Teclas de Funções

V1.0

<CODE>+<função> Programa Função

<ESC> Encerra / Máximo 15 caracteres

F1 = FUNÇÕES ^M

F2 = DIR ^M

F3 = STAT *.* ^M

F4 = ERA

F5 = PIP

F6 = TERM ^M

F7 = DIR B: ^M

F8 = STAT B: *.* ^M

F9 = DRINP ^M

F10 = DROUT ^M

Depreende-se que, para acionar um dos comandos listados, é suficiente pressionar a tecla correspondente.

Os comandos com terminação ^M (=RETURN) são prontamente executáveis, enquanto que os das tecla F4 e F5 devem receber as complementações necessárias antes de sua execução através do RETURN, que deverá ser dado através da tecla competente.

Qualquer das teclas F1 a F10 poderá ser reprogramada. Para tanto, o procedimento a adotar é o indicado a seguir:

- pressionar a tecla CODE simultaneamente com a tecla a ser reprogramada. Aparecerá em seguida no vídeo, após o cursor do sistema, o sinal = (igual), seguido do número da tecla;
- digitar então o comando desejado e pressionar em seguida a tecla ESC.

Para que o novo comando programado inclua o RETURN, a fim de ser de pronta execução, devem ser digitados após ele os caracteres ^M (sinal de potenciação e "m" maiúsculo), precedidos de um espaço, antes de ser pressionada a tecla ESC, que confirma a alteração.

A nova configuração dada à tecla é perdida quando o computador é desligado.

O PIP é o comando utilitário do HB-MCP que oferece o maior número de recursos e opções.

Sua finalidade é possibilitar a transferência entre dispositivos e arquivos e o processamento convencional destes.

Sua sintaxe é:

```
PIP nomepro destino=nomepro origem[,no-  
mepro origem][,nomepro origem]],no-  
mepro origem][opções]
```

ou PIP dispositivo=nomepro [opções]

ou PIP nomepro=dispositivo [opções]

ou PIP dispositivo=dispositivo [opções]

Para que seus múltiplos recursos sejam entendidos, é necessário que, antes, sejam identificados os dispositivos, palavras-chaves e opções válidos em seus comandos e reconhecidos pelo HB-MCP.

DISPOSITIVOS LÓGICOS:

CON: - Console, compreendendo na sua configuração normal teclado e vídeo, podendo este ser de 40 ou 80 colunas.

ENS: - Dispositivo de entrada serial de dados.

SDS: - Dispositivo de saída serial de dados.

LST: - Na configuração normal, dispositivo para impressão de dados em papel, constituído por impressora paralela.

DISPOSITIVOS FÍSICOS:

Para utilização com CON:

TV4: - Teclado e vídeo de 40 colunas.

TV8: - Teclado e vídeo de 80 colunas.

BAT: - O que estiver configurado em ENS: e LST:

TSS: - Teclado e o que estiver configurado com SDS:

Para utilização com ENS:

ESR: - Entrada serial normal.

TCL: - Teclado.

ES1: e ES2: - Tem a mesma função de ESR:

Para utilização com SDS:

SSR: - Saída serial normal.

SSE: - Saída serial com "eco" ao que estiver configurado em LST:.

SS1: e SS2: - Tem a mesma função de SSR:.

Para utilização com LST:

VD4: - Vídeo de 40 colunas.

VD8: - Vídeo de 80 colunas.

IMP: - Impressora paralela.

LSR: - Saída serial comum.

PALAVRAS-CHAVES:

(São especificadas com a mesma sintaxe dos dispositivos.)

NUL: - Envia 40 códigos de "nulo" (&H00) para o dispositivo configurado como de saída.

EOF: - Envia o caractere EOF (&H1A) para o dispositivo de saída.

INP: - Utilização especial avançada.

OUT: - Utilização especial avançada.

RDR: - Entrada com os protocolos de uma perfuradora de fita.

PUN: - Saída com os protocolos de uma perfuradora de fita.

OPÇÕES:

- A Copia apenas os arquivos que sofreram alterações.
- B Modo bloqueado. Os dados são considerados até receber um X-OFF (CTRL+S) do dispositivo origem. Utilizada na transferência de dados entre dispositivos de leitura/gravação contínua, como gravador cassete, dispositivo serial etc.
- Dn Elimina todos os caracteres situados além da coluna especificada por "n".
- E Exibe no vídeo todos os caracteres em operações de transferência através de outros dispositivos.
- F No processo de cópia de arquivos por impressora, elimina os caracteres &H0C (FORM FEED=CTRL+L), que ocasionam avanço para outra página (salto sobre o picoteado do papel).
- Gn Efetua leitura de arquivo armazenado por USER especificado por "n" para o USER corrente.
- H Transferência de arquivos no modo hexa-INTEL, com verificação da integridade dos dados neste formato.
- I Na transferência de arquivos no modo hexa-INTEL, ignora os registros tipo ":00". Aciona automaticamente a opção H.
- L Converte todos os caracteres alfabéticos para formato minúsculo, no processo de cópia de arquivos ou programas.
- N Adiciona indicador de linhas para o destino. A numeração começa em 1 e é incrementada de 1 em 1. Os números são seguidos de dois-pontos (:).

N2 O mesmo que N, não sendo os números seguidos de dois-pontos.

O No processo de cópia de programa (não de arquivo) é utilizada para indicar ao sistema que a existência no mesmo de códigos &H1A (EOF=CTRL+Z) deve ser ignorada para que a cópia seja completa. Se não for adotada, o processo é interrompido quando é encontrado tal código no programa.

Em cópias de arquivos, a opção "O" é adotada automaticamente pelo utilitário PIP.

Pn No processo de cópia por impressora, adiciona a cada "n" linhas um código &H0C (FORM FEED=CTRL+L) para que ocorra um avanço para outra página (salto do picoteado em formulário contínuo). Se n=1 ou não for especificado, é adotada a quantidade de 60 linhas.

Qs^Z Encerra a transferência de dados quando encontra uma cadeia de caracteres igual a "s" ("s" pode ser uma sucessão de caracteres quaisquer ou uma palavra.) Vide "Observação" mais abaixo.

R Permite efetuar cópias de arquivos com atributos .SYS, que, por vias normais, não são copiáveis.

Ss^Z Inicia a transferência de dados (copia) quando encontra uma cadeia de caracteres igual a "s" ("s" pode ser uma sucessão qualquer de caracteres ou uma palavra). Vide "Observação" abaixo.

Tn No processo de execução de cópia, quando encontra o caractere &H09 (TAB), converte a tabulação para "n" colunas.

U Nos processos de transferência, converte todos os caracteres para o formato maiúsculo.

V Nos processo de cópia, verifica se a gravação foi feita corretamente, relendo os registros copiados.

W No processo de cópia, mesmo tendo o arquivo destino atributo R/O, o comando será executado. Antes, porém, será solicitada confirmação através do vídeo.

Z Zera o "bit" 7 dos "bytes" em processo de transferência, ficando válidos apenas os caracteres da tabela ASCII, cujo número maior de código é 127.

Observação: quando PIP é usado no modo direto, todos os caracteres são convertidos para o formato maiúsculo e sem acentuação.

Assim, as opções "Q" e "S" devem ser usadas com o comando PIP no modo indireto, para que a interpretação da cadeia de caracteres seja feita corretamente pelo sistema operacional.

UTILIZAÇÃO DO COMANDO PIP:

PIP pode ser usado no modo direto. Exemplo:

```
A>PIP nomepro=nomepro
```

ou no modo indireto. Neste caso, é digitado apenas seu nome logo após o sinal do sistema de pronto para comando (A>) e pressionada a tecla RETURN, após o que surgirá na linha seguinte um asterisco (*), significando que o comando PIP aguarda digitação de parâmetros, para execução. Exemplo:

```
A>PIP
```

```
* parâmetros
```

No modo direto, após execução do comando, o sistema retorna novamente o cursor do sistema no modo de prontidão para comando.

No modo indireto, após execução do comando, retorna o asterisco, indicando que o comando PIP ainda está ativo e que aguarda digitação de novos parâmetros, para nova execução. Se uma nova execução não for desejada, deverão ser pressionadas simultaneamente as teclas

CONTROL e C

para retorno à condição normal de comandos. Exemplo:

```
A>PIP
```

```
* parâmetros
```

```
* parâmetros
```

```
* CONTROL+C
```

```
A>PIP
```

Os caracteres especiais * e ? são válidos no comando PIP.

EXEMPLOS DE UTILIZAÇÃO DO COMANDO PIP:

COPIAS DE PROGRAMAS OU ARQUIVOS:

Deve ser observado que a execução de cópias de programas ou arquivos sem alteração de nome só é possível se houver dois DISK DRIVES operando no sistema. Caso contrário, a execução de cópia de um mesmo programa ou arquivo só pode ser efetuada no mesmo disquete, dando-se nome diferente à cópia.

Dando-se o comando:

```
PIP B:=A:TEXTART1.BAS
```

ou

```
PIP B:TEXTART1.BAS=A:
```

será copiado o programa TEXTART1.BAS do disquete alojado no DISK DRIVE A: no disquete alojado no DISK DRIVE B:.

```
PIP A:=B:*. *[ROV]
```

Neste caso, todos os programas e arquivos do disquete em B: serão copiados no disquete em A:, inclusive programas com atributo .SYS. Os códigos de fim de arquivo existentes em programas (não arquivos) serão ignorados e será feita verificação de fidedignidade da gravação.

CONCATENAÇÃO DE ARQUIVOS:

```
PIP ARQ1=ARQ1,ARQ2
```

Com este comando, os dados dos ARQ1 e ARQ2 serão reunidos sob o nome ARQ1, e o ARQ2 continuará sem alteração.

```
PIP A:ARQ3=ARQ1[E],ARQ2[E]
```

Com esse comando, será criado o ARQ3 reunindo os dados dos ARQ1 e ARQ2, e estes também continuarão existindo sem alterações. A opção "E" realizará a exibição no vídeo dos caracteres em transferência.

PIP BOOK=B:CAPT1[FUE],CAPT2[FUE]

Com este comando, os arquivos CAPT1 e CAPT2 serão reunidos no arquivo BOOK, sendo removidos todos os caracteres FORM FEED (códigos indicadores de avanço para nova página); todos os caracteres alfabéticos serão convertidos para formato maiúsculo e a concatenação será mostrada no vídeo, isto é, o processo de transferência de caracteres.

PIP

*REUNE=HEX5[I],HEX6[LNP50]

O comando acima fará o arquivo REUNE receber primeiramente o arquivo HEX5 no formato INTEL, sendo eliminados registros do tipo :00, e em seguida o arquivo HEX6. Os caracteres alfabéticos deste serão convertidos para o formato minúsculo, suas linhas serão numeradas e será inserido um código de avanço de página a cada 50 linhas.

TRANSFERÊNCIA DE DADOS ENTRE ARQUIVOS E

DISPOSITIVOS E DISPOSITIVOS E ARQUIVOS:

PIP CON:=CHAPT1.ARQ

Com o comando acima, os dados do arquivo CHAPT1.ARQ serão exibidos no vídeo.

PIP TEXT02.ARQ=CON:

Dado o comando acima, o que for digitado em seguida será gravado no arquivo TEXT02.ARQ quando forem pressionadas as teclas CONTROL e Z.

PIP TEXT5=ENS:[B:],EOF:

Com este comando, todos os caracteres que entrarem através da interface serial (ou cartucho serial) serão gravados no arquivo TEXT5, até que, pela mesma via, entre um código X-OFF (CTRL+S). Após entrar um código EOF pela serial, o mesmo será enviado para o arquivo TEXT5.

PIP

*CON:LETTER[Singênuo^ZQesperança^ZT12]

O comando acima fará com que o conteúdo do arquivo de nome LETTER seja exibido no vídeo a partir da cadeia de caracteres "ingênuo" até a palavra "esperança", com tabulações de 12 colunas onde encontrar o código correspondente a TAB.

Se o comando for dado com PIP no modo direto, e o texto do arquivo estiver gravado em letras minúsculas, o comando não surtirá efeito. Como observado antes, na forma direta o comando PIP converte os caracteres digitados nos seus parâmetros para o formato maiúsculo, ocorrendo, em consequência, desencontro no processo de busca.

No comando acima, se, em vez de CON: for indicado IMP:, o conteúdo do arquivo será enviado para a impressora, também a partir da palavra "ingênuo" até a palavra "esperança".

Todas as operações efetuadas por PIP são monitoradas no vídeo com mensagens de apoio ao sistema e de ocorrências.

Tais mensagens estão reunidas na "TABELA DE MENSAGENS DO HB-MCP", no Apêndice, no final do livro.

PECULIARIDADES DO UTILITÁRIO PIP:

Por razões de segurança, quando efetua a cópia de um programa ou arquivo, antes de ser encerrado o processo o mesmo é armazenado provisoriamente num arquivo de caráter temporário sob o nome especificado no comando, porém com extensão na forma .\$\$\$. Apenas depois de executada a cópia sem problemas é que o programa ou arquivo recebe o nome e extensão definitivos, conforme especificados no comando.

Assim, poderá ocorrer que o programa ou arquivo com extensão .\$\$\$ permaneça também no diretório do disquete depois de encerrado o processo, por razões diversas. Em tal caso, depois de constatada no disquete a existência da cópia equivalente e definitiva, será necessário deletá-lo através do comando ERA.

Normalmente, tal providência não se faz necessária.

É comando utilizado para alterar ou mudar o nome de programas ou arquivos em disquete.

Sua sintaxe é:

REN [d:]nomepro novo=nomepro antigo

d: deverá ser especificado se houver dois DISK DRIVES em operação no sistema e o disquete com o programa a ser processado estiver alojado na unidade não corrente.

Os caracteres * e ? não são válidos para uso com este comando.

Se já existir no disquete programa ou arquivo com nome idêntico ao novo nome especificado no comando, o sistema operacional emitirá a seguinte mensagem através do vídeo:

Já existe

e cancelará o comando.

Se o arquivo ou programa indicado não existir no disquete, a mensagem emitida será:

Nenhum arquivo

Através deste comando são salvas - ou gravadas - em disquete partições do conteúdo da AREA DE PROGRAMAS TRANSIENTES (TPA), área essa criada na RAM do computador pelo HB-MCP para armazenamento de arquivos ou programas em processamento e que ocupa toda a região compreendida entre os endereços 256 (&H100) e 52503 (&HD0FF).

Cada partição dessa área corresponde a 256 "bytes" e é denominada "página".

A sintaxe do comando é:

SAVE n₀ [d:]nomepro

em que n₀ especifica a quantidade de "páginas" ou partições da citada área a serem gravadas no disquete, começando sempre pela primeira, no endereço 256.

O número máximo de páginas é 255.

Por exemplo, dado o comando

SAVE 10 PROVMEM.ARQ

todo o conteúdo dos "bytes" entre os endereços 256 e 2815 será gravado no disquete sob o nome PROVMEM.ARQ.

Constitui conteúdo da área de programas transientes o programa ou arquivo por último processado, podendo ser comando interno ou externo.

STAT é outro comando utilitário do HB-MCP de múltiplos recursos.

Fornecer informações sobre o DISK DRIVE, disquete, programas e arquivos.

Mostra no vídeo a configuração atual dos dispositivos lógicos com os físicos e permite reconfigurar seu alinhamento ou combinação.

Converte programas ou arquivos normais para a condição R/O, que permite efetuar somente leitura, protegendo-os contra escrita acidental, e retorna-os à condição R/W, que permite leitura e escrita, fazendo a mesma coisa com o(s) DISK DRIVE(S).

Converte também programas com atributo .SYS para programas normais e vice-versa.

Sua sintaxe é:

STAT [parâmetros]

Os dispositivos lógicos e físicos válidos para STAT são os mesmos descritos para o comando PIP.

Os caracteres especiais * e ? são aceitos por STAT.

EXEMPLOS DE COMANDOS:

Dado simplesmente, isto é, sem parâmetros,

STAT

exibirá no vídeo os seguintes dados:

STAT

Informações sobre o disco

V1.0

A: R/W, Espaço 256K

A: indica o DISK DRIVE corrente.

R/W significa READ/WRITE, indicando que o disquete acionado pode ser lido ou gravado.

Espaço 256K é a indicação do espaço disponível no disquete.

Se houver dois DISK DRIVES operando no sistema, serão exibidas as informações citadas sobre cada um deles.

O comando

STAT TEXTART1.BAS

fornecerá as seguintes informações sobre o programa especificado e o espaço disponível no disquete:

Regs	Bytes	Ext	Atr
------	-------	-----	-----

48	6k	1	R/W A:TEXTART1.BAS
----	----	---	--------------------

Bytes restantes em A: 246K

Regs indica a quantidade de registros utilizados pelo programa. Cada registro ocupa 128 "bytes".

Bytes indica a quantidade de "bytes" alocados para o programa, sempre em múltiplos de 2. No HB-MCP, a alocação mínima para programas ou arquivos, ou frações destes, é de 2 "kbytes". Mesmo que um programa utilize somente 200 "bytes", a área alocada para ele no disquete será de 2 "kbytes".

Ext indica a quantidade de "extensões" ocupadas pelo programa. Cada extensão corresponde a 16 "kbytes", ou 8 blocos de dados, sendo que cada bloco corresponde a 16 registros de 128 "bytes". (Vide comando FORMAT.)

Atr indica o atributo do programa ou arquivo, podendo ser

R/W, para leitura e escrita, R/O, apenas para leitura.
Bytes restantes em A: indicação do espaço ainda disponível
no disquete alojado no DISK DRIVE especificado.

O comando

STAT FIPES.ARQ \$\$

tem o mesmo efeito do comando anterior, porém informa também o tamanho
potencial de um arquivo de acesso aleatório.

O comando

STAT *.*

fornece informações sobre todos os programas e arquivos armazenados no
disquete alojado no DISK DRIVE corrente.

STAT FICHAPES.ARQ \$R/O

converte o arquivo especificado para leitura somente.

As conversões permitidas para programas e arquivos e suas indica-
ções são:

\$R/O converte somente para leitura.

\$R/W converte para leitura e escrita.

\$SYS converte para sistema.

\$DIR converte para normal.

Após uma conversão, é emitida mensagem de confirmação pelo siste-
ma operacional.

Programas e arquivos de sistema(s) (atributo .SYS) não são lista-
dos pelos comandos DIR e STAT.

O comando:

STAT A:=R/O

converte o DISK DRIVE A temporariamente operacional apenas para leitura. Mensagem confirmando conversão é emitida pelo sistema no vídeo.

O comando:

STAT A:DSK:

lista no vídeo todas as características do disquete alojado em A:.

O comando:

STATUSR:

informa qual é o USER ativo e todos que compartilham o disquete.

STATVAL:

exibe no vídeo um resumo dos comandos permitidos pelo STAT.

STATDEV:

informa pelo vídeo qual é a configuração corrente dos dispositivos lógicos e físicos, com o seguinte quadro:

STAT

Informações sobre o disco

V1.0

CON: é TV4

ENS: é ESR:

SDS: é SSR:

LST: é IMP:

O comando:

STAT dispositivo lógico=dispositivo físico

recombina um dispositivo lógico com um dispositivo físico, em conformidade com as configurações possíveis descritas no comando PIP.

Da mesma forma que ocorre com os outros comandos do HB-MCP, todas as execuções comandadas por STAT são monitoradas no vídeo, com emissão de mensagens de apoio ao sistema e orientação ao operador.

Tais mensagens estão reunidas na "TABELA DE MENSAGENS DO HB-MCP", no Apêndice, no final do livro.

É um comando utilitário que gera execução automática e sequencial de comandos do HB-MCP reunidos sob a forma de arquivo, nos moldes do "comando coletivo" ou "processamento em lote" do MSX DOS.

Sua sintaxe é:

SUBMIT [d:]nomepro

em que nomepro é o nome sob o qual o arquivo foi gravado no disquete.

Para que seja reconhecido pelo sistema operacional como um arquivo de comandos - para execução automática - seu nome deve ter a extensão .SUB, como indicado no exemplo que segue:

NOMEPRO.SUB

A sequência de execução dos comandos pode ser interrompida pressionando-se simultaneamente as teclas

CONTROL e C

entre a execução de um e outro comando.

Segue um exemplo de um arquivo de comandos, gravado sob o nome

AFERIDOR.SUB

Comandos incluídos:

```
DIR
STAT ARQ1.TXT
TYPE TEXTART1.ARQ
PIP ARQ2.TXT=ARQ1.TXT
STAT ARQ2.TXT
TYPE ARQ2.TXT
```

Para que todos os comandos acima sejam executados automaticamente e na sequência em que foram inseridos, constituindo um "comando coletivo", bastará dar o comando:

SUBMIT AFERIDOR

Note-se que, em analogia com arquivos de comandos (.COM), não deve figurar no comando de execução a extensão .SUB.

CRIAÇÃO DE UM ARQUIVO DE COMANDOS:

Utiliza-se o comando PIP para se criar um arquivo de comandos, na seguinte forma:

A>PIP nomepro.SUB=CON:

ou

A>PIP

* nomepro.SUB=CON:

digitando-se em seguida os comandos na ordem desejada.

Depois de digitado cada comando, ou linha de comando, deve-se teclar RETURN. Isto fará com que o cursor volte para o início da mesma linha, sem apagá-la.

Para mudar o cursor de linha e digitar outro comando, será necessário pressionar simultaneamente as teclas CONTROL e J.

Para encerrar a inserção de comandos no arquivo, devem ser pressionadas simultaneamente as teclas CONTROL e Z e, em seguida, CONTROL e C, após o que o arquivo será gravado automaticamente no disquete.

Não deve haver erro durante a digitação dos comandos, porque nessa modalidade o comando PIP não permite correções no seu "buffer", embora elas sejam efetuadas na monitoração de vídeo.

OBSERVAÇÃO:

Pode-se criar um arquivo coletivo de modo mais prático através do sistema operacional MSX DOS. (Consulte item 10.4 - "COMANDO COLETIVO OU PROCESSAMENTO EM LOTE".)

Depois de criado e gravado em disquete através do MSX DOS, o arquivo de comandos pode ser transferido facilmente para um disquete do sistema HB-MCP, com auxílio do comando COPDOS deste.

Obviamente, e não obstante ser criado através do MSX DOS, os comandos que comporão o arquivo deverão ser os do HB-MCP. É imprescindível que o nome do arquivo tenha a extensão .SUB.

A finalidade do utilitário SYSGEN é copiar em disquete já formatado os programas que constituem o sistema operacional para disco HB-MCP e que são carregados automaticamente no computador quando este é ligado, estando no DISK DRIVE corrente tal disquete.

Sua sintaxe é simplesmente o próprio nome, opcionalmente seguido da indicação do DISK DRIVE:

SYSGEN [d:]

A indicação do DISK DRIVE a ser acionado deverá ser feita somente se houver duas unidades operando no sistema.

Os programas que constituem os comandos internos do HB-MCP só podem ser copiados de um disquete para outro através do utilitário SYSGEN, ou pelo utilitário PIP com a opção R quando houver dois DISK DRIVES operando.

Poderão eventualmente ser copiados através do utilitário BACKUP, mas com considerável desvantagem no tempo consumido.

Os programas que constituem propriamente o sistema HB-MCP residem obrigatoriamente nas três primeiras trilhas do disquete e apenas estas são copiadas através de SYSGEN. Nelas estão incluídos os comandos internos do HB-MCP. O seu conteúdo não é listado por DIR e STAT.

Os comandos externos ou temporários, que são em verdade programas utilitários - também reconhecidos como comandos utilitários - não são copiados por SYSGEN, que, por sinal, também faz parte deles.

Tais utilitários podem residir em quaisquer áreas do disquete e são copiados de um para outro através de COPY ou PIP, que pertencem à mesma categoria.

Todo o processo executado por SYSGEN é monitorado no vídeo, através do qual são indicadas ao operador as trocas que devem ser efetuadas no alojamento dos disquetes envolvidos, quando apenas um DISK DRIVE operar no sistema.

A seguir são reproduzidas as mensagens geradas no vídeo durante a execução de SYSGEN:

SYSGEN

Gera Sistema

V1.0

Coloque disquete no drive A:

para receber sistema

Tecla <RETURN>

Outra cópia (S/N) ?

Coloque disquete com sistema

Tecla <RETURN>

TERM altera a configuração de caracteres de controle de vídeo.

Normalmente, o sistema está configurado para o padrão VT-52, podendo ser mudado para o padrão ADM-3A, permitindo assim que programas escritos para este padrão possam ser processados pelo HB-MCP sem necessidade de outras providências que não a de selecionar o padrão desejado através da simples escolha das opções oferecidas por este utilitário.

A seleção é feita através do "menu" reproduzido a seguir:

TERM

Seleciona Terminal

V1.0

(A) Terminal VT-52

(B) Terminal ADM-3A

(Z) Não altera

Opção:

Através deste comando é exibido no vídeo o conteúdo de um arquivo (não programa) gravado no disquete.

Sua sintaxe é:

TYPE [d:]nomepro

Se d: não for especificado, será acionado o DISK DRIVE corrente.

A sequência de exibição dos dados no vídeo pode ser sustada pressionando-se simultaneamente as teclas

CONTROL e S.

Pressionando-se em seguida qualquer tecla, a exibição prossegue.

Se for pressionada isoladamente qualquer tecla durante a exibição de dados, o processo será abortado.

Os dados exibidos no vídeo podem também ser enviados à impressora se, antes de ser dado o comando TYPE, forem pressionadas ao mesmo tempo as teclas

CONTROL e P

que ativam a função de duplicação de dados por aquele periférico. (Recorra ao item 11.3, "FUNÇÕES DE CONTROLE POR CARACTERE NO HB-MCP".)

Em tal caso, convirá configurar previamente a saída de dados para a impressora pelo padrão que reproduz corretamente as palavras em língua portuguesa, com acentos e "ç", o que é feito facilmente através do comando DROUT do próprio HB-MCP.

O comando USER permite que um disquete seja utilizado com até 16 partições, sem divisões fixas e espaços predeterminados, sendo válido apenas o limite correspondente à capacidade máxima normal de armazenamento do disquete.

Sua sintaxe é:

USER no

em que n^o significa a partição, podendo variar de 0 a 15.

Ao ser ativado o HB-MCP, o USER que se torna automaticamente corrente é o 0, e seu número não é visualizado com o cursor.

Para ser mudado para outro número, basta digitar após o cursor do sistema em prontidão o nome do comando e o número desejado. Por exemplo, se for digitado:

USER 7

e pressionada a tecla RETURN, o cursor do sistema passará a ser precedido pelo número 7 (7A>), e todas as operações de entrada e saída no e do disquete serão efetuadas na partição de número 7 do mesmo, significando que, nessa condição, são e serão acessíveis apenas os programas e arquivos que foram ou que serão inseridos no disquete sob controle do USER 7. Dado um comando DIR nesta condição, por exemplo, serão listados no vídeo apenas os programas e arquivos pertencentes ao USER 7.

Para retornar ao USER 0, basta digitar:

USER 0

e pressionar em seguida a tecla RETURN.

É importante observar que a partição de número 7, ou qualquer outra, passa a existir apenas a partir do momento em que é inserido no disquete qualquer programa ou arquivo sob comando do USER especificado

e terá o tamanho ou extensão da soma de programas e arquivos inseridos sob o mesmo número.

Os comandos residentes no HB-MCP continuam normalmente válidos e os externos só serão disponíveis para os USERS em cujas partições sejam também gravados, o que pode ser feito com o procedimento indicado a seguir:

- digita-se o comando utilitário a ser duplicado para outro USER.

Vamos fazê-lo com STAT.COM:

STAT

- depois de teclar RETURN, pressiona-se simultaneamente as teclas

CONTROL e C.

(Com esse procedimento o utilitário STAT.COM passa para a área de programas transientes (TPA) e o cursor volta à prontidão.)

- digita-se:

USER 1

- tecla-se RETURN. O cursor passa para a condição 1A>.

- digita-se em seguida:

SAVE 20 STAT.COM

Quando o cursor voltar à prontidão, o utilitário STAT estará disponível ao USER 1, o que pode ser constatado dando-se o comando DIR.

(O número 20 no comando acima indica o número de "páginas" a gravar da TPA e é calculado com base na extensão do programa STAT, obtida através de seu próprio comando. Como toma 40 registros de 128 "bytes", tal extensão corresponde a 20 páginas de 256 "bytes" da TPA.)

Os demais comandos externos são transferidos para o USER 1 ou outro qualquer pelo mesmo método.

Depois de transferido o comando utilitário PIP.COM para um deter-

minado USER, quaisquer programas ou arquivos podem ser transferidos ou copiados para o mesmo, usando-se a opção "G" de PIP. Por exemplo, para gravar no USER 7 o arquivo CHAPT1.ARQ existente no USER 0, adota-se o seguinte procedimento:

```
A>USER 7      (RETURN)
```

```
7A>PIP CHAPT1.ARQ=CHAPT1.ARQ[G0]      (RETURN)
```

O comando USER pode ser muito útil para agrupar seletivamente arquivos e programas em partições diferentes do disquete, conferindo ao sistema maior agilização e otimização.

A finalidade do comando XSUB é permitir a execução automática total de comandos agrupados em arquivos com atributo .SUB quando de tais arquivos fizerem parte comandos de leitura do "buffer" do teclado.

Quando é detectado num arquivo tal comando de leitura, o XSUB intercepta sua execução e, em vez de aguardar a entrada de dados através do teclado, introduz os dados que foram inseridos no elenco de comandos do arquivo com atributo .SUB, em linha seguinte à sua figuração no mesmo.

O comando XSUB e os dados a serem inseridos no elenco de comandos de execução automática do arquivo coletivo devem ser digitados na exata ordem de sequência em que devem ser executados.

11.3 - FUNÇÕES DE CONTROLE POR

CARACTERE NO HB-MCP

Da mesma forma que no MSX DOS, a função de controle por caractere do HB-MCP é uma função que afeta a linha de comando ou o comando propriamente dito, alterando-o ou complementando-o em sua digitação ou em sua execução.

Qualquer função de controle por caractere é ativada pressionando-se a tecla CONTROL (ou CTRL) simultaneamente com a tecla cujo caractere é designado no sistema como de controle.

No HB-MCP tais caracteres são:

C, E, H, I, J, M, P, R, S, U, X, Z.

Por exemplo, em certos comandos, pressionando-se a um só tempo as teclas

CONTROL e Z

é produzido um código &H1A, correspondente a EOF, indicando ao sistema operacional "fim de arquivo" ou denotando "fim de entrada de dados".

As funções de controle por caractere poderão ser ativadas sempre que o cursor do sistema estiver em prontidão para comando ou durante a digitação de um comando.

Para facilidade de acesso e consultas, todas as funções desta categoria são descritas na "TABELA DE FUNÇÕES DE CONTROLE POR CARACTERE DO HB-MCP", no Apêndice, no final do livro.

11.4 - OCORRÊNCIAS DE ERROS EM OPERAÇÕES

DO HB-MCP E MENSAGENS DO SISTEMA

Durante a execução de um comando, quando ocorre alguma modalidade de erro de leitura ou de gravação, o sistema operacional emite através do vídeo uma mensagem correspondente.

Quando o erro é irre recuperável, a mensagem induz o operador a decidir e indicar que alternativa adotar dentre as oferecidas na forma de opções, como na mensagem reproduzida a seguir:

Erro no drive d: R/O

^C Cancela, RETURN Ignora R/O

ou como nesta outra:

Erro no drive d: Setor Ruim

^C Cancela, RETURN repete,

Ignora com qualquer tecla

ou, quando não há alternativa:

Erro no drive d: Seleção

Aperte uma tecla

Em tais casos, geralmente o erro é superado quando ignorado e é pressionada a tecla que ocasiona a repetição de execução do comando.

Há casos de erros irre recuperáveis, felizmente raros, quando a única solução é desligar o computador e o sistema operacional e começar tudo novamente, tendo que, possivelmente, regravar um ou outro programa ou mesmo sofrer a perda de um ou diversos, sendo necessário colocar fora de uso o disquete envolvido. Essa é, aliás, a principal razão que reforça a conveniência de ter sempre programas e arquivos - principalmente os mais importantes - duplicados em disquetes de reserva ou mesmo em fitas cassete.

Há também os erros do operador, como falhas na digitação de nomes de comandos ou de programas e arquivos, esquecimento de recolocar no DISK DRIVE um disquete etc.

Além das mensagens referentes a erros, há as mensagens de simples

apoio ao sistema, visando completa integração com o operador.

Para facilidade de acesso e consultas, tais mensagens são descritas com seus respectivos significados na "TABELA DE MENSAGENS DO HB-MCP", no Apêndice, no final do livro.

11.5 - PECULIARIDADES DO HB-MCP

TROCA DE DISQUETES:

Já vimos que o HB-MCP só pode ser ativado através de disquete que contenha os programas de sistema que o constituem.

Normalmente, no mesmo disquete que contém os programas do sistema há diversos outros programas, inclusive os que constituem os comandos externos ou utilitários do HB-MCP, de forma que, ao ser efetuada sua leitura, o sistema operacional memoriza uma série de informações sobre o mesmo, a fim de poder processá-lo corretamente.

Por inerência do sistema, tais informações são específicas sobre cada disquete em particular.

Assim, se for efetuada troca de disquetes quando o sistema se encontrar em prontidão para comando - portanto não durante a execução de um comando que exija esse procedimento -, os programas de sistema devem ser recarregados na memória do computador através do novo disquete, a fim de que o sistema operacional capte e organize em sua memória as informações necessárias correspondentes a ele.

Para tanto, depois de alojado o novo disquete, basta pressionar simultaneamente as teclas

CONTROL e C.

Se tal não for feito, o sistema operacional iniciará a execução de determinados comandos baseado nas informações referentes ao disquete anterior, ocorrendo então um tipo de erro cuja mensagem nem sempre condiz perfeitamente com a situação, deixando o operador confuso.

NOMES DE PROGRAMAS E ARQUIVOS:

Por ser um programa totalmente compatível com o famoso CP/M, do qual é originário e, portanto, mais afeito ao trato e processamento de palavras sem acentuação gráfica nem "ç" com cedilha, a estruturação do HB-MCP pode esbarrar, e mesmo tropeçar, em certas circunstâncias, com essas particularidades da língua portuguesa.

Por tais razões, e a título de providência, será conveniente evitar, tanto quanto possível, a adoção de caracteres acentuados graficamente e "ç" nos nomes de programas e arquivos processados pelo sistema, não obstante a versão nacional produzida pela EPCOM Equipamentos Eletrônicos Ltda. ter sido munida de recursos que visaram evitar problemas dessa ordem.

A P Ê N D I C E

Este apêndice reúne assuntos de interesse sobre a matéria tratada neste livro, bem como as tabelas referenciadas ao longo do texto.

Foi colocada no final do livro com o propósito de propiciar maior facilidade de acesso para consultas.

Sabemos que a linguagem BASIC foi criada para facilitar o entrosamento entre o computador e o seu operador.

Contudo, enquanto o BASIC é generosamente acessível ao operador - sendo quase o seu próprio linguajar comum, em inglês -, ele não significa nada para a máquina propriamente, a qual, para funcionar dentro dos padrões que conhecemos, deve receber instruções em códigos numéricos, os quais, por sua vez, são transformados em "pulsos elétricos" de natureza apropriada a fazê-la trabalhar segundo tais padrões.

Assim, quando um programa em BASIC é introduzido no computador, ele passa obrigatoriamente por um programa interpretador residente em sua memória permanente e é convertido em códigos numéricos.

Para que tal ação possa ocorrer, todas as palavras do BASIC que representam comandos, instruções e funções, bem como as letras do alfabeto, os números e os símbolos em geral são codificados em números, geralmente expressos no sistema hexadecimal.

Letras do alfabeto, números e símbolos em geral são reunidos em tabelas-padrões. A mais difundida delas, de âmbito internacional, é a TABELA ASCII. Outra, de padrão nacional, é a BRASC, da ABNT. Os computadores do padrão MSX também têm sua tabela.

As letras do alfabeto - com exceção das acentuadas da língua portuguesa e do caractere "ç" -, os números e alguns símbolos têm os mesmos códigos numéricos nas três tabelas.

As palavras reservadas ao sistema operacional do computador - comandos, instruções e funções - têm códigos numéricos próprios do padrão ou linha a que pertence o produto.

Baseados na tabela do padrão MSX, se digitarmos a frase:

```
PRINT "CASA"
```

e teclarmos RETURN em seguida, o que vai efetivamente para o seu centro de processamento é a seguinte sucessão de códigos hexadecimais:

91	20	22	43	41	53	41	22	00
		"	C	A	S	A	"	
		CÓD. DE ESPAÇO						
		CÓD. DE PRINT						
		CÓD. DE FIM DE LINHA						

sendo que o último código (00) é colocado automaticamente pelo sistema operacional do computador quando teclamos RETURN, significando fim de instrução.

Se a instrução não for precedida por um número - caracterizando-a como linha de programa - 10, por exemplo, ela é simplesmente executada e apagada ou descartada, isto é, não permanece na memória da máquina.

Se lhe for dado um número de linha de programa, receberá do sistema operacional outros códigos, correspondentes ao número de linha de instrução e à caracterização como linha de programa, e permanecerá na memória do computador como uma instrução definitiva - enquanto não for eliminada ou substituída deliberadamente, ou o computador não for desligado - e poderá ser executada sempre que um comando apropriado, como RUN ou GOTO, por exemplo, for dado.

Uma linha de instrução nessa configuração, ou seja:

10 PRINT "CASA"

será armazenada na memória do computador através da seguinte sucessão de códigos hexadecimais:

0E	80	0A	00	91	20	22	43	41	53	41	22	00
				"	C	A	S	A	"			
				CÓD. DE ESPAÇO								
				CÓD. DE PRINT								
		Nº DA LINHA										
END. PRÓX. LINHA DE INSTR. NA RAM												
CÓDIGO DE FIM DE LINHA												

Esse é o tipo de instrução em BASIC que pode ser transferido da memória do computador para uma memória auxiliar: disquete, fita casete ou cartucho.

Tal transferência - normalmente referida como gravação - pode ser feita em duas modalidades: compacta e plena, denominadas:

- GRAVAÇÃO EM FORMATO BINÁRIO

- GRAVAÇÃO EM FORMATO ASCII

respectivamente.

No formato BINÁRIO, o conteúdo de memória do computador - correspondente a uma instrução ou programa em BASIC - é gravado exatamente na forma em que está armazenado, como no exemplo citado, sem passar de volta pelo programa interpretador, ou seja: não é decodificado e não é retornado para gravação na forma em que foi digitado.

No formato ASCII, o mesmo conteúdo de memória passa de volta pelo programa interpretador e é decodificado para ser gravado exatamente com os códigos correspondentes à forma em que foi digitado, ou seja:

31	30	20	50	52	49	4E	54	20	22	43	41	53	41	22	0D	0A
1	0		P	R	I	N	T		"	C	A	S	A	"		
CÓDIGOS DE ESPAÇO										CÓDIGOS DE FIM DE LINHA						

utilizando quatro - 4 - códigos a mais por linha do que no formato binário. Os códigos 0D e 0A são colocados automaticamente pelo sistema operacional, indicando fim de instrução e avanço de linha.

Se considerarmos uma só linha de instrução, a diferença não é relevante. Todavia, se for considerado um programa de 11264 "bytes", por exemplo, no formato binário, ele ocupará no disquete espaço correspondente a 13294 "bytes" se for gravado no formato ASCII.

A diferença na quantidade de "bytes" poderá variar ligeiramente de um programa para outro, dependentemente da disposição e formato das instruções, já que são as palavras representativas delas, dos comandos e das funções que sofrem compactação na RAM, quando são convertidas em

códigos numéricos, com a finalidade de indicar ao sistema operacional sua função.

Há a considerar, outrossim, que o tempo requerido para gravar um programa em BASIC no formato ASCII é sensivelmente mais longo do que no formato binário. Por exemplo, o mesmo programa acima citado demanda aproximadamente cinco (5) segundos para ser gravado em formato binário e requer aproximadamente vinte e cinco segundos (25) para ser gravado em formato ASCII.

Teoricamente, a gravação em formato ASCII é mais confiável.

Mas pondo de lado o fato de que no formato ASCII qualquer programa é processável pelo comando MERGE e que há diferenças no espaço ocupado no disquete e no tempo de gravação, em nossas experiências não conseguimos ainda perceber qualquer diferença na confiabilidade de ambos os processos considerados.

Da mesma forma que ocorre quando um programa em BASIC é gravado no formato ASCII, ao ser dado um comando LIST, por exemplo, o programa existente na memória do computador passa de volta pelo programa interpretador e é retornado ao vídeo na mesma forma em que foi digitado.

As gravações de programas em LINGUAGEM DE MÁQUINA obedecem a critério diferente.

Como vimos no comando BSAVE, este só é executável quando especifica os endereços inicial e final da área de memória do computador em que se localiza o programa em linguagem de máquina a ser gravado.

Tal tipo de programa normalmente é armazenado em área protegida, acima do topo da RAM, depois de este ter sido deslocado para baixo por meio da instrução CLEAR.

Localizado nessa área, um programa em linguagem de máquina fica fora do alcance geral dos comandos do BASIC, podendo ser acessado apenas através das funções USR e DEFUSR. Sua gravação em disquete corresponde exatamente à disposição em que seus códigos componentes são mantidos na área protegida da RAM do computador.

SEQUENCIAL OU ALEATÓRIO

Há dois tipos de arquivos de dados que podem ser armazenados em disquete. Normalmente, são denominados:

ARQUIVO SEQUENCIAL

e ARQUIVO DE ACESSO ALEATÓRIO OU DIRETO

Em arquivos convencionais de escritório, dados são armazenados em pastas. Quando se deseja examinar o conteúdo de uma delas, verifica-se no índice do arquivo sua localização e ela é simplesmente apanhada.

Em arquivos em disquete, dados são armazenados, por processo semelhante ao de gravação de som em fitas cassete, em espaços físicos da superfície de material plástico que o constitui - superfície essa revestida de fina película de partículas magnéticas - e são denominados "registros".

O registro de um arquivo de dados é, pois, equivalente a uma pasta de arquivo convencional de escritório.

Quando se deseja examinar um determinado registro, damos instruções ao computador, através de um programa gerenciador do arquivo - um arquivo em disquete ou fita só subsiste através de um programa que, de fato, permite criá-lo, mantê-lo e manipulá-lo - e o registro é buscado no arquivo pelo sistema operacional, podendo seus dados ser exibidos no vídeo ou passados para o papel através de impressora.

Em ARQUIVO SEQUENCIAL, essa busca é feita pelo sistema operacional sempre a partir do começo do arquivo, isto é, ela sempre é iniciada pelo primeiro registro e prossegue até localizar aquele cujos dados se enquadrem nos parâmetros especificados na instrução de busca. Por exemplo, se determinados dados requisitados se encontrarem no registro que foi gravado na 123ª posição, o sistema operacional pesquisará todos os 122 registros precedentes, comparará os dados de cada um com os dos parâmetros especificados para busca e se deterá apenas naquela po-

sição, ficando à disposição do vídeo ou da impressora os dados correspondentes, através das instruções PRINT ou LPRINT.

Assim, em arquivo sequencial não há meios de acessar diretamente um registro, como o 1230, por exemplo, ainda que se saiba que os dados desejados se encontram nele.

Em ARQUIVO DE ACESSO ALEATÓRIO OU DIRETO, a busca pode ser dirigida diretamente a qualquer registro. Por exemplo, se se sabe que determinados dados desejados se encontram no registro gravado na posição 456a do arquivo, indicamos como parâmetro na instrução de busca apenas esse número e o sistema operacional se dirige diretamente àquele registro, tornando imediatamente disponíveis para o vídeo ou para a impressora os dados correspondentes.

Em síntese, essas são as características básicas de cada modalidade de arquivo em disquete.

Quanto à conveniência de escolha de um ou outro tipo de arquivo, diversos outros fatores devem ser considerados e cada caso deverá ser examinado em função dos mesmos. Citaremos a seguir os principais:

TAMANHO OU EXTENSÃO DO ARQUIVO

- Em ARQUIVO SEQUENCIAL a extensão do registro pode variar num mesmo arquivo, não sendo necessário predeterminá-la. A extensão máxima admissível para cada registro é de 256 "bytes".

- Em ARQUIVO DE ACESSO ALEATÓRIO a dimensão dos registros não pode variar e tem que ser predeterminada. Quando não utilizados os espaços predeterminados para o(s) seu(s) campo(s), o sistema operacional os preenche automaticamente com o código de "espaço em branco". A extensão máxima admissível para cada registro é também de 256 "bytes".

CORREÇÃO OU ALTERAÇÃO DE DADOS DOS REGISTROS

- Em ARQUIVO SEQUENCIAL é impossível efetuar correções por via direta. A única forma - em verdade pouco prática - é transferir todos os registros para variáveis indexadas, dentro dos limi-

tes da memória do computador, efetuar as correções necessárias e regravar todo o arquivo, o que limita o seu tamanho ao da memória livre do computador.

- Em ARQUIVO DE ACESSO ALEATÓRIO a correção de qualquer registro pode ser efetuada de maneira direta e imediata, bastando recuperá-lo do disquete para a memória do computador, efetuar as alterações desejadas e regravá-lo na mesma posição que ocupava no disquete, o que é feito automaticamente pelo sistema operacional.

Esta característica do ARQUIVO DE ACESSO ALEATÓRIO torna seu tamanho praticamente ilimitado, já que lhe permite comportar até 4.294.967.295 registros.

ABERTURA E FECHAMENTO DOS ARQUIVOS

- O ARQUIVO SEQUENCIAL requer abertura e fechamento a cada acesso ou modalidade de operação.

- O ARQUIVO DE ACESSO ALEATÓRIO pode ser aberto uma única vez para todo o processamento e para qualquer modalidade de operação, devendo ser fechado somente quando é encerrado o trabalho com ele.

CRIAÇÃO E PROCESSAMENTO DE ARQUIVOS EM DISQUETE

Como já mencionamos, arquivos em disquetes ou fitas só subsistem através de PROGRAMAS GERENCIADORES. Isto significa que devem ser criados e processados por programas próprios, dos quais são praticamente inseparáveis.

Vamos apresentar a seguir dois PROGRAMAS GERENCIADORES que permitem criação e processamento dos dois tipos descritos de arquivos em disquete, a fim de que sejam elucidados através dos mesmos os diversos comandos, instruções e funções utilizados por eles e descritos em nosso livro. Começemos com ARQUIVO SEQUENCIAL, na página seguinte:

ARQUIVO SEQUENCIAL

Comandos, instruções e funções utilizados:

CLOSE #, EOF, INPUT #, LINE INPUT #, LOC,
LOF, MAXFILES, OPEN, PRINT #, PRINT # USING

PROGRAMA GERENCIADOR DE ARQUIVO SEQUENCIAL

```
10 '##### PROGRAMA MALDIRLS.BAS
20 SCREEN 0: WIDTH 40: KEYOFF
30 CLEAR 2000: MAXFILES=2
40 '##### MENU
50 CLS: LOCATE 5,5: PRINT "MALA DIRETA"
60 PRINT TAB(5) STRING$(11,45)
70 PRINT,,TAB(10)"1 - INICIA REGISTROS"
80 PRINT,,TAB(10)"2 - CONTINUA REGISTROS"
90 PRINT,,TAB(10)"3 - BUSCA POR NOME"
100 PRINT,,TAB(10)"4 - BUSCA POR REGISTRO"
110 PRINT,,TAB(10)"5 - LISTA REG. NO VIDEO"
120 PRINT,,TAB(10)"6 - IMPRIME ETIQUETAS"
130 PRINT,,TAB(10)"7 - ENCERRA OPERAÇÕES"
140 R$=INPUT$(1)
150 ON VAL(R$) GOTO 180,330,420,550,670,750,810
160 GOTO 140
170 '##### ABERTURA DO ARQUIVO, ENTRADA E GRA
    VAÇÃO DE DADOS
180 CLS:OPEN "A:MALDIRLS.ARQ" FOR OUTPUT AS #1:C=1
190 IF C=1 THEN PRINT,,SPC(7)"Comece a digitar os
    dados.":FOR T=1 TO 500:NEXT T:GOTO 230
200 IF C>1 THEN PRINT,,SPC(10)"Outra entrada? (s/n
    )":R$=INPUT$(1)
210 IF R$="S" OR R$="s" THEN 230
220 CLOSE #1:GOTO 40
```

```

230 CLS:PRINT,,TAB(12)"REGISTRO Nº";C: R=C
240 PRINT:INPUT"NOME: ";N$
250 PRINT:LINEINPUT"ENDEREÇO: ";E$
260 PRINT:INPUT"CEP: ";P$
270 PRINT:LINEINPUT"CIDADE, ESTADO: ";C$
280 PRINT,,TAB(6)"Registra (r) ou corrige (c)?"
290 R$=INPUT$(1):IF R$="C" OR R$="c" THEN 230 ELSE
    IF R$="R" OR R$="r" THEN 310
300 GOTO 290
310 PRINT #1,R:PRINT #1,N$:PRINT #1,E$:PRINT #1,P$
    :PRINT #1,C$
320 C=C+1:GOTO 190
330 '##### CONTINUA REGISTROS
340 CLS:OPEN "A:MALDIR1S.ARQ" FOR INPUT AS #1:C=1
350 IF EOF(1)=-1 THEN 380
360 INPUT #1,R:INPUT #1,N$:LINEINPUT #1,E$:INPUT #
    1,P$:LINEINPUT #1,C$
370 C=C+1:GOTO 350
380 CLOSE #1
390 OPEN "A:MALDIR1S.ARQ" FOR APPEND AS #1
400 CLS:PRINT,, " Digite os novos registros."
410 FOR T=1 TO 500:NEXT T:GOTO 230
420 '##### PROCURA POR NOME
430 CLS:PRINT:PRINT SPC(23):PRINT"NOME A PROCURAR:
    ":PRINT:INPUT X$
440 OPEN "A:MALDIR1S.ARQ" FOR INPUT AS #1:C=1
450 IF EOF(1)=-1 THEN 480
460 INPUT #1,R:INPUT #1,N$:LINEINPUT #1,E$:INPUT #
    1,P$:LINEINPUT #1,C$
470 IF N$=X$ THEN 610 ELSE C=C+1:GOTO 450
480 PRINT:PRINT,,TAB(8)"Não consta no arquivo.":GO
    TO 540
490 PRINT:PRINT,,TAB(9)"DADOS SOBRE O ARQUIVO:"
500 PRINT:PRINT,, "Extensão do arquivo: ";LOF(1);"by
    tes."
510 L=INT(LOC(1)/256):L1=L*256
520 PRINT,, "Partições usadas: ";L;"+";LOF(1)-L1;"by
    tes."

```

```

530 PRINT: PRINT" (Cada partição: 256 bytes.)"
540 CLOSE:PRINT:PRINT,, "Pressione uma tecla para v
oltar ao MENU.":R$=INPUT$(1):GOTO 40
550 '##### PROCURA POR REGISTRO
560 CLS:PRINT:INPUT"NUMERO DO REGISTRO A BUSCAR";B
570 OPEN "A:MALDIR1S.ARQ" FOR INPUT AS #1:C=1
580 IF EOF(1)=-1 THEN 480
590 INPUT #1,R:INPUT #1,N$:LINEINPUT #1,E$:INPUT #
1,P$:LINEINPUT #1,C$
600 IF B=R THEN 610 ELSE C=C+1:GOTO 580
610 '##### EXIBE REGISTRO NO VIDEO
620 CLS:LOCATE 0,7:PRINT"REGISTRO Nº";R
630 PRINT,, "NOME: ";N$
640 PRINT,, "END.: ";E$
650 PRINT,, "CEP ";P$;" - ";C$
660 GOTO 540
670 '##### LISTA REGISTROS EM TV E IMPRESSORA
680 CLS:LOCATE 0,10:PRINT"Tecle STOP para deter/co
ntinuar exibição";,, " Pressione qualquer tecla pa
ra iniciar.":R$=INPUT$(1)
690 OPEN "A:MALDIR1S.ARQ" FOR INPUT AS #1:OPEN "CR
T:" AS #2
700 IF EOF(1)=-1 THEN 490
710 INPUT #1,C:INPUT #1,N$:LINEINPUT #1,E$:INPUT #
1,P$:LINEINPUT #1,C$
720 PRINT #2,:PRINT #2,:PRINT #2,SPC(2);C
730 PRINT #2,N$:PRINT #2,E$:PRINT #2,P$;" - ";C$
740 C=C+1:GOTO 700
750 '##### IMPRIME ETIQUETAS
760 CLS:LOCATE 0,10:PRINT"   APRONTE A IMPRESSORA
E TECLE 'RETURN'PARA COMEÇAR OU QUALQUER TECLA PAR
A SUS-PENDER E VOLTAR AO MENU":R$=INPUT$(1)
770 IF ASC(R$)-13 THEN 780 ELSE 40
780 OPEN "A:MALDIR1S.ARQ" FOR INPUT AS #1
790 OPEN "LPT:" AS #2
800 GOTO 700
810 CLOSE
820 CLS:LOCATE 0,18:PRINT"NÃO SE ESQUEÇA DE DESLIC
AR OS APARELHOS!":END

```

ARQUIVO DE ACESSO ALEATÓRIO

Comandos, instruções e funções utilizados:

CLOSE #, CVD, CVI, CVS, EOF, FIELD, GET #, LOC, LOF
LSET, MAXFILES, MKD\$, MKI\$, MKS\$, OPEN, PUT, RSET

PROGRAMA GERENCIADOR DE ARQUIVO ALEATÓRIO

```
10 '##### PROGRAMA MALDIR2D.BAS
20 SCREEN 0:WIDTH 40:KEYOFF
30 CLEAR 2000:MAXFILES=2
40 '##### ABERTURA DO ARQUIVO
50 OPEN "A:MALDIR2D.ARQ" AS #1 LEN=85
60 R=LOF(1)/85+1
70 FIELD #1,27 AS NMS,27 AS ENS,5 AS CPS,24 AS CIS
  ,2 AS ES$
80 '##### MENU
90 CLS:LOCATE 5,5:PRINT"MALA DIRETA"
100 PRINT TAB(5)STRING$(11,45)
110 PRINT,,TAB(10)"1 - EFETUA REGISTROS"
120 PRINT,,TAB(10)"2 - BUSCA POR NOME
      C/OPÇÃO P/ALTERAR)"
130 PRINT,,TAB(10)"3 - BUSCA POR REGISTRO
      C/OPÇÃO P/ALTERAR)"
140 PRINT,,TAB(10)"4 - LISTA REG. NO VIDEO"
150 PRINT,,TAB(10)"5 - IMPRIME ETIQUETAS"
160 PRINT,,TAB(10)"6 - ENCERRA OPERAÇÕES"
170 R$=INPUT$(1)
180 ON VAL(R$) GOTO 200,350,480,720,790,920
190 GOTO 170
200 '##### ENTRADA E GRAVAÇÃO DE DADOS
210 CLS:PRINT,,TAB(12)"REGISTRO Nº:";R
220 PRINT:INPUT "NOME(27):";N$
230 PRINT:LINE INPUT "ENDER.(27):";E$
```

```

240 PRINT:INPUT "CEP:";P$
250 PRINT:INPUT "CIDADE(24): ";C$
260 PRINT:INPUT "ESTADO(2): ";T$
270 LSET NM$=N$:LSET EN$=E$:LSET CP$=P$:LSET CI$=C
$:LSET ES$=T$
280 PRINT:PRINT TAB(7)"Registra(r) ou corrige(c)?"
290 R$=INPUT$(1):IF R$="C" OR R$="c" THEN 200 ELSE
  IF R$="R" OR R$="r" THEN 310
300 GOTO 290
310 PUT #1,R:R=R+1
320 PRINT:PRINTTAB(10)"Outro registro? (s/n)"
330 R$=INPUT$(1):IF R$="S" OR R$="s" THEN 200 ELSE
  IF R$="N" OR R$="n" THEN 80
340 GOTO 330
350 '##### PROCURA POR NOME
360 IF R<2 THEN 890
370 CLS:PRINT:PRINT SPC(23):PRINT"NOME A PROCURAR:
":PRINT:INPUT N$
380 FOR C=1 TO R-1
390 GET #1,C
400 IF N$=LEFT$(NM$,LEN(N$)) THEN 440
410 NEXT C
420 PRINT:PRINT,,TAB(9)"Não consta no arquivo."
430 GOTO 870
440 GOSUB 660
450 PRINT:PRINT " Altera(a), cancela(c), volta men
u(v)?"
460 R$=INPUT$(1):IF R$="A" OR R$="a" THEN 570 ELSE
  IF R$="C" OR R$="c" THEN 530 ELSE IF R$="V" OR R$
="v" THEN 80
470 GOTO 460
480 '##### PROCURA POR REGISTRO
490 IF R<2 THEN 890
500 CLS:PRINT:PRINT" Qual o nº de registro a busc
ar";:C=0:INPUT C:IFC=>R THEN PRINT,,TAB(6)"O últim
o nº do arquivo é:";R-1:GOTO 870
510 GET #1,C:GOSUB 660
520 GOTO 450
530 '##### CANCELA REGISTRO

```



```

540 LSET NM$="":LSET EN$="":LSET CP$="":LSET CI$="
":LSET ES$="":PUT #1,C
550 PRINT:PRINTTAB(10)"Registro cancelado."
560 GOTO 870
570 '##### ALTERA REGISTRO
580 PRINT:INPUT"Novo nome(27)";N$:LSET NM$=N$
590 LINE INPUT "Endereço(27)";E$:LSET EN$=E$
600 INPUT "CEP";P$:LSET CP$=P$
610 INPUT "Cidade(24)";C$:LSET CI$=C$
620 INPUT "Estado(2)";T$:LSET ES$=T$
630 PUT #1,C
640 PRINT:PRINT TAB(8)"Alterações já efetuadas!"
650 GOTO 870
660 '##### IMPRESSÃO NO VIDEO
670 PRINT:PRINT,,SPC(2);C
680 PRINT: PRINT NM$
690 PRINT EN$
700 PRINT CP$;SPC(3);CI$;ES$
710 RETURN
720 '##### LISTA REGISTROS NO VIDEO
730 IF R<2 THEN 890
740 CLS:LOCATE 0,10:PRINT"Tecla STOP para deter/co
ntinuar exibição";,, " Pressione qualquer tecla par
a iniciar.":R$=INPUT$(1)
750 FOR C=1 TO R-1
760 GET #1,C:GOSUB 660
770 NEXT C
780 GOTO 870
790 '##### IMPRIME ETIQUETAS
800 IF R<2 THEN 890
810 CLS:LOCATE 0,10:PRINT:PRINT"   APRONTE A IMPRE
SSORA E TECLE 'RETURN'PARA COMEÇAR OU QUALQUER TEC
LA PARA SUS-PENDER E VOLTAR AO MENU":R$=INPUT$(1)
820 IF ASC(R$)=13 THEN 830 ELSE 80
830 OPEN "LPT:" AS #2
840 FOR C=1 TO R-1
850 GET #1,C:PRINT#2,:PRINT#2,:PRINT#2,SPC(2);C:PR
INT#2,NM$:PRINT#2,EN$:PRINT#2,CP$;SPC(3);CI$;ES$

```

```

860 NEXT C:CLOSE #2
MENU." :R$=INPUT$(1)
880 GOTO 80
890 CLS:LOCATE 3,12:PRINT"AINDA NÃO HÁ REGISTROS N
O ARQUIVO!":TIME=0
900 IF TIME<120 THEN 900
910 GOTO 80
920 CLOSE:CLS:LOCATE 0,11:PRINT"NÃO SE ESQUEÇA DE
DESLIGAR OS APARELHOS!":END

```

Observação: os dois programas apresentados destinam-se à mesma finalidade, um sendo versão do outro, propositadamente, nas modalidades destinadas a:

ARQUIVO SEQUENCIAL

e a

ARQUIVO DE ACESSO ALEATÓRIO.

Visamos assim propiciar ao leitor a possibilidade de analisar pelo método de comparação as diferenças essenciais entre os dois tipos de programas gerenciadores das duas modalidades de arquivos, bem como, através de sua utilização prática na criação e no processamento de arquivos, poder perceber claramente as características de cada um com relação a recursos para acesso aos registros, alterações, velocidade, capacidade de armazenamento etc.

A.3 - TABELA DAS CHAVES ESPECIAIS

DE EDIÇÃO DO MSX DOS

NOME DA CHAVE / TECLAS DA CHAVE

FUNÇÃO DA CHAVE

COPY1 **CURSOR P/DIR. ou CONTROL+**

Edita um a um todos os caracteres.

COPYUP **SELECT ou CONTROL+Xe**

Edita todos os caracteres até o especificado "e".

COPYALL **CURSOR P/BAIXO ou CONTROL+_**

Edita de uma só vez todos os caracteres.

SKIP1 **DEL**

Não edita, isto é, pula um caractere por vez.

SKIPUP **CLS ou CONTROL+Le**

Não edita ou pula os caracteres até o especificado "e".

VOID **CURSOR P/CIMA ou ESCAPE**

ou CONTROL+^ ou CONTROL+U ou CONTROL+E

Invalida a linha de comando digitada e mantém o "template".

BS **CURSOR P/ESQ. ou CONTROL+H ou CONTROL+J**

Apaga o último caractere digitado.

INSERT

INS ou CONTROL+L

Ativa ou desativa o modo inserção na linha de comando.

NEW LINE

HOME ou CONTROL+K

Passa a linha de comando para o "template".

Observações: na descrição da função da chave, a palavra "edita" significa "reproduz na linha de comando".

"Linha de comando" é a linha onde se posiciona o cursor em prontidão para comando e na qual este é digitado.

"Template" é uma área especial de memória em que sempre é armazenada a última linha de comando dado ao sistema operacional.

Consulte o item 10.5, "CHAVES ESPECIAIS DE EDIÇÃO", sobre outros detalhes relativos a esta tabela.

A.4 - TABELA DAS FUNÇÕES DE CONTROLE

POR CARACTERE DO MSX DOS

CONTROLE / FUNÇÃO

CONTROL+C

Cancela ou aborta um comando em execução.

CONTROL+H

Anula o último caractere da linha de comando.

CONTROL+J

Muda o cursor para a linha de baixo.

CONTROL+M

O mesmo que RETURN ("carriage return" = retorno de carro).

CONTROL+N

Cancela o "eco" na impressora, isto é, deixa de enviar para ela o que é exibido no vídeo.

CONTROL+P

Ativa o sistema "eco", que passa a enviar para a impressora tudo o que é exibido no vídeo.

CONTROL+S

Suspende ou imobiliza a exibição de dados no vídeo. O controle é desativado pressionando-se qualquer tecla.

Observação: para que a função descrita seja ativada, as duas teclas indicadas devem ser pressionadas juntamente.

A.5 - TABELA DAS FUNÇÕES DE CONTROLE

POR CARACTERE DO HB-MCP

CONTROLE / FUNÇÃO

CONTROL+C

Recarrega no computador os programas de sistema do HB-MCP.

CONTROL+E

Muda o cursor de linha de comando, sem afetar sua continuidade.

CONTROL+H

Volta e apaga caracteres da linha de comando. O mesmo que BS.

CONTROL+I

Move o cursor para a próxima tabulação.

CONTROL+J

Termina a linha de comando atual. O mesmo que RETURN e CONTROL+M.

Quando ativado o utilitário PIP, na modalidade PIP nomepro=CON:, CONTROL+J é o único controle que muda o cursor de linha, para fins de digitação.

CONTROL+M

O mesmo que RETURN ou CR ("carriage return" = retorno de carro).

CONTROL+P

Ativa/desativa o sistema "eco", que envia para a impressora tudo o que é exibido no vídeo.

CONTROL+R

Repete ou exhibe novamente a linha de comando corrente.

CONTROL+S

Imobiliza/continua exibição de dados no vídeo. Uma tecla qualquer pressionada avulsamente cancela a exibição dos dados.

CONTROL+U

Cancela a linha de comando corrente, reiniciando outra sem apagar a cancelada.

CONTROL+X

Cancela e apaga a linha de comando corrente e reinicia outra.

CONTROL+Z

Finaliza entrada de dados. Produz código de fim de arquivo (EOF).

Observação: para que a função descrita seja válida, as duas teclas indicadas devem ser pressionadas simultaneamente.

A.6 - TABELA DE MENSAGENS DO MSX DISC BASIC

Nº DO CÓDIGO / MENSAGEM (PORTUGUÊS/INGLÊS) / MOTIVO

50 Campo maior

Field overflow

A quantidade de "bytes" determinada pela instrução FIELD excede a prevista por LEN para cada registro do arquivo de acesso direto.

51 Erro interno

Internal error

Ocorrência de mal funcionamento do BASIC.

52 Número do arquivo

Bad file number

Referência feita por comando ou instrução a número de arquivo não aberto ou fora de faixa.

53 Arquivo não existe

File not found

Referência feita por comando ou instrução a arquivo que não existe no disquete.

54 Arquivo aberto

File already open

Comando OPEN feito no modo FOR OUTPUT para arquivo sequencial já aberto, ou uso do comando KILL para arquivo aberto.

55 Fim do arquivo

Input past end

Instrução INPUT executada depois de todos os registros de um arquivo sequencial já terem sido lidos ou recuperados, ou executada em relação a um arquivo nulo ou vazio. A função EOF é usada para detectar "fim de arquivo" e evitar esse tipo de erro.

56 Nome do arquivo

Bad file name

Referência errada feita por comando ou instrução a nome de arquivo, geralmente por falha na digitação.

57 Comando direto no arquivo

Direct statement in file

Comando LOAD utilizado para carregamento de arquivo.

58 Arquivo sequencial

Sequential I/O only

Comandos GET e PUT usados em arquivo sequencial.

59 Falta OPEN

File not open

Comando feito em relação a arquivo não aberto.

60 Erro de disco

Bad allocation table

Utilização de disquete não formatado ou com falha na formatação.

61 Modo errado

Bad file mode

Uso indevido de PUT, GET ou LOF com arquivo sequencial, ou LOAD para carregar arquivo em formato ASC II. Tentativa de executar um comando OPEN em arquivo sequencial, em modo que não seja FOR INPUT, FOR OUTPUT, FOR APPEND, ou tentativa de adoção destes modos em arquivo de acesso direto.

62 Drive errado

Bad drive name

Especificação de DISK DRIVE inválida.

63 Erro no setor

Sector error

Existência de falha num setor do disquete.

64 Arquivo ainda aberto

File still open

Permanência indevida de arquivo aberto.

65 Arquivo já existe

File already exists

Comando NAME especificando nome de programa ou arquivo idêntico a um já existente no disquete.

66 Disco cheio

Disk full

Esgotamento da capacidade de armazenamento do disquete.

67 Diretório cheio

Too many files

Esgotamento de espaço no diretório do disquete, impossibilitando cadastrar novos programas ou arquivos.

68 Disco protegido

Disk write protected

Comando feito para cópia ou gravação de programa, arquivo ou registro em disquete protegido contra gravação.

69 Erro I/O

Disk I/O error

Ocorrência de erro durante uma operação de entrada/saída no e do disquete (gravação/reprodução), sem possibilidade de correção.

70 Falta disco

Disk offline

DISK DRIVE acionado sem disquete.

71 Modo inválido

Rename across disk

Comando RENAME dado de modo inválido.

Observação: as mensagens de números 50 a 59 são as mesmas do BASIC.

A.7 - TABELA DE MENSAGENS DO MSX DOS

MENSAGEM (PORTUGUÊS/INGLÊS) / MOTIVO

MENSAGENS DE APOIO:

Aperte uma tecla

Strike any key

Parada na execução sequencial de um COMANDO COLETIVO, ao ser encontrado o comando PAUSE.

Encerra auto comando (S/N)?

Terminate batch file (Y/N)?

Interrupção na execução de um COMANDO COLETIVO, ao ser ativada a função de controle por caractere CONTROL+C.

Insira disco c/arquivo batch

Aperte uma tecla

Insert disk with batch file and

strike any key when ready

O disquete que contém o COMANDO COLETIVO em processamento foi retirado do DISK DRIVE operante.

Insira disco no drive d:

e pressione qualquer tecla

Insert diskette for drive d:

and strike a key when ready

Quando o sistema opera com um só DISK DRIVE, necessidade de efetuar a troca de disquetes para continuidade de processamento.

Tem certeza (S/N)?

Are you sure (Y/N)?

Confirmar se o comando DEL *.* dado (que elimina do disquete todos os programas ou arquivos) deve mesmo ser executado.

MENSAGENS DE ERRO:

Coloque disco com sistema

e aperte uma tecla

Insert DOS disk in default drive and

strike any key when ready

Foi normalmente carregado do disquete para o computador o primeiro programa integrante do MSX DOS e não é encontrado no disquete o segundo programa que compõe o sistema, imprescindível, pois é o programa de comandos do mesmo.

Comando inexistente

Bad command or file name

Digitação errada de nome de comando ou de programa ou arquivo, ou indicação de inexistência dos mesmos no disquete.

Comando perdeu-se antes da cópia

Content of destination lost before copy

Na concatenação de arquivos através do comando COPY, existência de arquivo destino entre os arquivos fontes.

Data inválida

Invalid date

Digitação errada ou fora das faixas previstas pelo sistema.

Disco protegido para escrita em d:

Cancela, Repete ou Ignora?

Write protect error writing drive d:

Abort, Retry, Ignore?

Tentativa de copiar programa(s) ou arquivo(s) em disquete protegido contra escrita ou gravação.

Drive inválido

Invalid drive specification

Especificação de DISK DRIVE inválida.

Erro de disco em d:

Bad FAT, drive d:

Ocorrência de erro no disquete.

Erro de disco p/escrita (ou p/leitura) em d:

Cancela, Repete ou Ignora?

Disk error reading (ou writing) drive d:

Abort, Retry, Ignore?

Ocorrência de erro no processo de gravação ou de leitura no e do disquete.

Erro na criação do arquivo

File creation error

Impossibilidade de inserir o arquivo ou programa no disquete, por falta de espaço nele ou no diretório.

Erro na escrita

Write error

Ocorrência de erro no processo de gravação no disquete.

Erro na troca de nomes

Rename error

Existência no disco de nome igual ao designado num comando REN.

Espaço insuficiente no disco

Insufficient disk space

Insuficiência de espaço no disquete para gravar programa ou arquivo através do comando COPY.

Hora inválida

Digite nova hora

Invalid time

Enter new time

Horário digitado errado ou fora das faixas previstas pelo sistema operacional, quando existe relógio interno no computador.

Não encontrou arquivo

File not found

Comando dado com nome de programa ou de arquivo não existente no disquete.

Não pode ser copiado nele mesmo

File cannot be copied onto itself

Tentativa de copiar no mesmo disquete um programa ou arquivo sem alterar-lhe o nome.

Não pronto p/escrita (ou p/leitura) em d:

Cancela, Repete ou Ignora?

Not ready error writing (ou reading) drive d:

Abort, Retry, Ignore?

Comando dado para gravação ou leitura sem haver disquete no DISK DRIVE corrente.

Não tem relógio

Comando TIME dado em computador MSX que não possui relógio interno, caso das primeiras versões dos computadores MSX brasileiros.

Parâmetro inválido

Invalid parameter

Indicação de parâmetro inválido para o comando MODE.

Programa muito longo

Program too big to fit in memory

Tentativa de processar um programa cuja extensão é maior do que a memória disponível no computador.

A.8 - TABELA DE MENSAGENS DO HB-MCP

MENSAGEM / MOTIVO OU PROVIDÊNCIA A TOMAR

Aperte uma tecla

Continuar execução de um comando ou exibição de dados no vídeo.

Arquivo existe. Elimina (S/N) ?

Existe arquivo ou programa com o mesmo nome no disquete destino.

Arquivo não encontrado

Não existe o arquivo ou programa no disquete, ou seu nome foi digitado com erro.

Assinalamento inválido

Especificação de dispositivo inválida por sintaxe ou por configuração indevida.

Caracter CTRL inválido

Utilização inválida de caractere de controle.

Coloque disquete com sistema

Disquete com sistema deve ser colocado no DISK DRIVE A:. Teclar RETURN em seguida para retornar ao HB-MCP.

Coloque disquete com sistema e

Tecla <RETURN>

Necessidade de recarregar o sistema HB-MCP no computador.

Coloque disquete destino e

Tecle <RETURN>

Alojar no DISK DRIVE o disquete que vai receber a(s) cópia(s).

Coloque disquete origem e

Tecle <RETURN>

Alojar no DISK DRIVE o disquete com o(s) arquivo(s) a copiar.

Coloque disquete destino em d:

Coloque disquete origem em d:

Tecle <RETURN>

Orientar troca de disquetes na execução de cópias.

Comando inválido !

Erro de sintaxe ou especificação inválida de DRIVE ou de disco.

Comando muito longo

Comando excedendo a capacidade de uma linha.

Copia todos (S/N) ?

Confirmar ou não execução de cópia de todos os programas e arquivos de um disquete.

Delimitador inválido

Uso de separador de dispositivos não válido no comando.

Registro é R/O. Elimina (S/N) ?

Arquivo destino protegido contra escrita. "S" elimina o atributo R/O e efetua a cópia.

Disco não reconhecido

Coloque disco HB-DOS e

Tecla <RETURN>

O disco exigido deve ter formato HB-DOS para receber cópia.

Disco/Diretório cheio

Falta de espaço no disquete ou no seu diretório.

Entrada configurável

O dispositivo designado como de entrada deve ser substituído por outro de entrada, pois depende dele para ter configuração correta.

EOF inesperado. ^Z p/encerrar

Encontrado indicador de fim de arquivo fora de posição. Pressionar CONTROL+Z para encerrar processo.

ERRO

Continua (S/N) ?

Ocorrência de algum tipo de erro. "N" cancela o comando. "S" continua o processo, ignorando o erro encontrado.

Erro de CHECKSUM

^Z encerra

Erro de rotina interna. CTRL+Z encerra o processo. RETURN continua, ignorando o erro.

Erro de ou na gravação

Gravação com erro ou disquete protegido contra escrita.

Erro de ou na leitura

Leitura impossibilitada por ocorrência de algum erro.

Erro de verificação !

Disquete não formatado corretamente.

Erro na formatação !

Existência de erro ou defeito na formatação do disquete.

Erro na linha xxx:

Linha com erro em arquivo de comandos com atributo .SUB.

Erro no arquivo destino

Cópia não completada por problemas no arquivo destino. Provável falta de espaço no disquete ou em seu diretório.

Erro no disco

Impossibilidade de gravar por falta de espaço no disquete ou em seu diretório ou por existência de defeito em algum setor.

Escolha opção:

Uma das opções descritas no "menu" deve ser escolhida.

Excesso de arquivos

Capacidade de classificação dos arquivos para exibição dos dados foi esgotada.

Excesso de comandos

Impossibilidade de criar arquivo com atributo .SUB por excesso de comandos e falta de espaço na RAM.

Falta memória

Excesso de comandos através do utilitário BATCH esgotando capacidade da RAM.

Fim não encontrado

Não foi encontrada a cadeia de caracteres indicada na opção "Q" do comando PIP, para encerrar a cópia.

Formato inválido

Não foi identificado pelo sistema operacional o comando dado.

Indicador de arquivo inválido

Sintaxe incorreta na especificação do nome do arquivo.

Início não encontrado

Não foi encontrada a cadeia de caracteres especificada na opção "S" do comando PIP, para início de cópia.

Já existe

Já existe no disquete programa ou arquivo com o mesmo nome especificado para a nova gravação ou para alteração.

Nenhum arquivo

Inexistência do ou de programa no disquete.

Não achou arquivo

O arquivo especificado no comando não existe no disquete.

Não fechou arquivo

Impossibilidade de fechar o arquivo por falta de espaço no disquete ou no seu diretório.

Não leu

Leitura especificada não efetuada. Erro de leitura ou problema no arquivo.

Não pode enviar

O dispositivo de saída não está preparado para processar dado.

Não pode receber

O dispositivo de entrada não está preparado para processar dado.

Não é dispositivo de entrada

O dispositivo indicado como de origem não é de entrada.

Não é dispositivo de saída

O dispositivo especificado para enviar dados não é de saída.

Outra cópia (S/N) ?

Confirmar ou não execução de outra cópia.

Parâmetro inválido

Parâmetro digitado com erro ou inválido.

Pode gravar no disco (S/N) ?

Confirmação solicitada pelo comando AUTOLD para gravar ou desgravar comando(s) autoexecutante(s).

Saída configurável

O dispositivo especificado como de saída deve ser substituído por outro de saída, pois depende dele para ter configuração correta.

Sem espaço

Disquete com espaço esgotado. Não pode receber novas gravações.

Sem espaço no diretório

Todas as entradas do diretório já foram usadas.

Separador inválido

O caractere usado como separador de nome de arquivo ou de dispositivo não é válido.

.SUB

Inexistência do arquivo SUBMIT especificado no comando.

Todos (S/N) ?

Solicitação de confirmação para eliminar todos os programas e arquivos do disquete, após um comando ERA *.*.

USER inválido

Tentativa de ler arquivo de um USER inexistente.

Versão inválida

Uso indevido de STAT com outra versão de sistema operacional.

XSUB ativado

XSUB continua ativo. CTRL+Z desativa.

XSUB já está ativado

XSUB já havia sido carregado.

A.9 - TABELA PARA USO DO TECLADO

DO EXPERT COM O HB-MCP

PARA DIGITAR:

USAR A TECLA:

?	-----	:
/	-----	;
:	-----	>
;	-----	<
"	-----	^
a	-----	"
*	-----	'
^	-----	)
\	-----	{
'	-----	}
..	-----	[
}	-----	]
[	-----	^
>	-----	?
<	-----	/
^	-----	~
^	-----	^

As teclas dos demais caracteres são correspondentes.

MAPA DE SEQUÊNCIA E SIGNIFICADO DOS "BYTES"

(DISC BASIC e MSX DOS)

ATRIBUTOS DO ARQUIVO										HORA DATA		
NOME DO ARQUIVO										PRIMEIRO GRUPO DE UM ARQUIVO		TAMANHO DO ARQUIVO
.	.	.	.	.	.	.	.	.	.	.	.	.

BIBLIOGRAFIA

BIBLIOTECA PÚBLICA DO PARANÁ

NÃO DANIFIQUE ESTA ETIQUETA



30 243 653



INTRODUCCIÓN AL MSX-DISK BASIC

SONY

MANUAL DE INSTALAÇÃO (VERSÃO PRELIMINAR)

9410 AP/B/D

ELEBRA INFORMÁTICA S/A.

MSX-DOS CP/M-80

Technical Information

MICROSOFT CORPORATION



MSX-DOS USER'S GUIDE

MSX-DOS COMMAND GUIDE

MSX DISK BASIC REFERENTE GUIDE

Publicações sem fonte citada.

SISTEMA OPERACIONAL HB-MCP

Versão 1.0 JUL/1986

EDUARDO REZENDE

EPCOM EQUIPAMENTOS ELETRÔNICOS LTDA.

OUTROS LIVROS NA ÁREA

- Avalon* — *O livro vermelho do MSX*
- Burd* — *MSX - Guia do Operador - Comandos Básicos*
- Burd* — *MSX - Jogos. Volumes I a III*
- Burd* — *Simulações no MSX*
- Bussab* — *MSX Música*
- Carvalho* — *Assembler para o MSX*
- Hoffman* — *MSX - Guia do Usuário*
- Marriot* — *MSX - O meu Primeiro Livro*
- Sato* — *MSX - Guia do Programador. Volumes 1 e 2*